

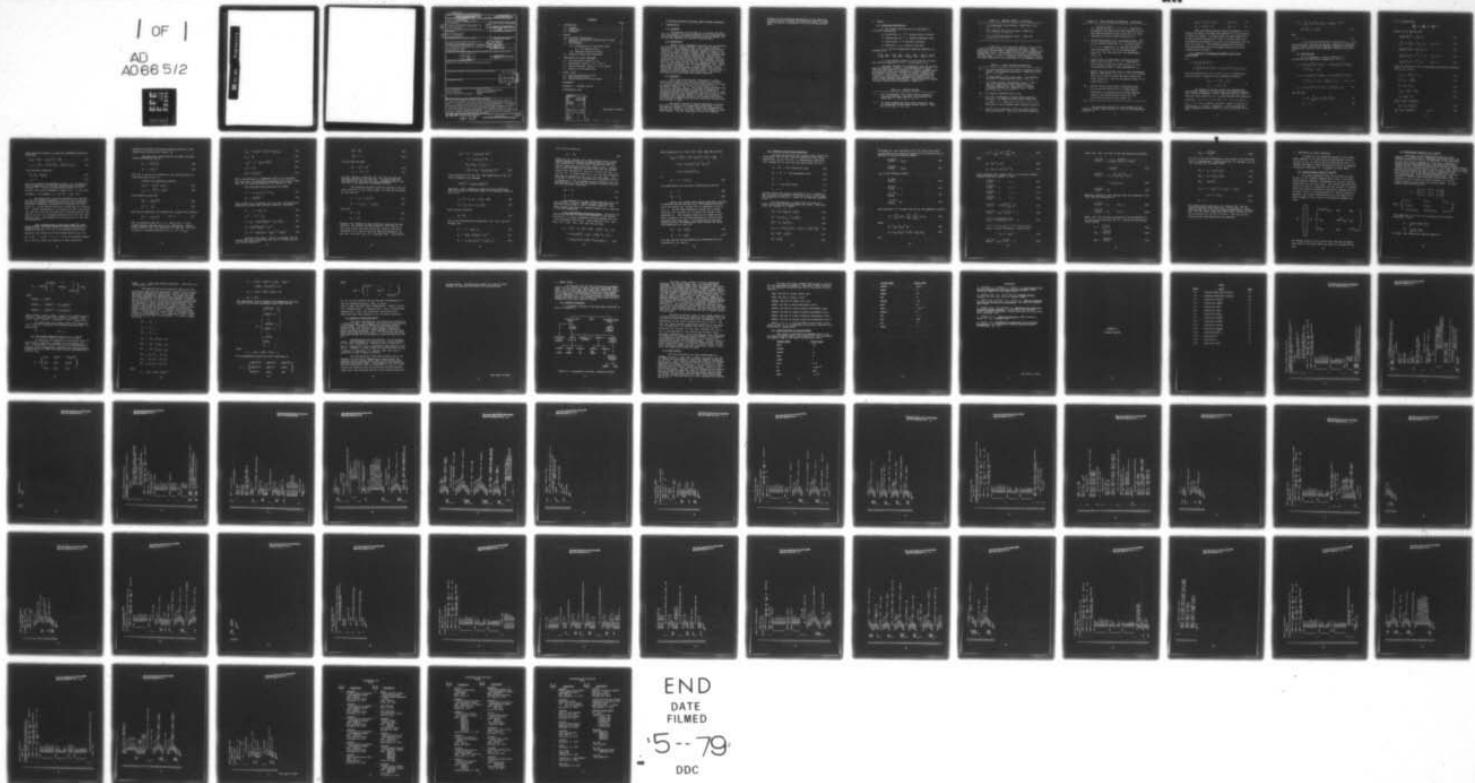
AD-A066 512

ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY ABERDEEN PROV--ETC F/G 17/9
A COMPUTER PROGRAM FOR DOUBLE SWEEP OPTIMAL SMOOTHING.(U)
JAN 79 R A SCHEDER

UNCLASSIFIED

R A S C H E D E R
AMSA-TR-246

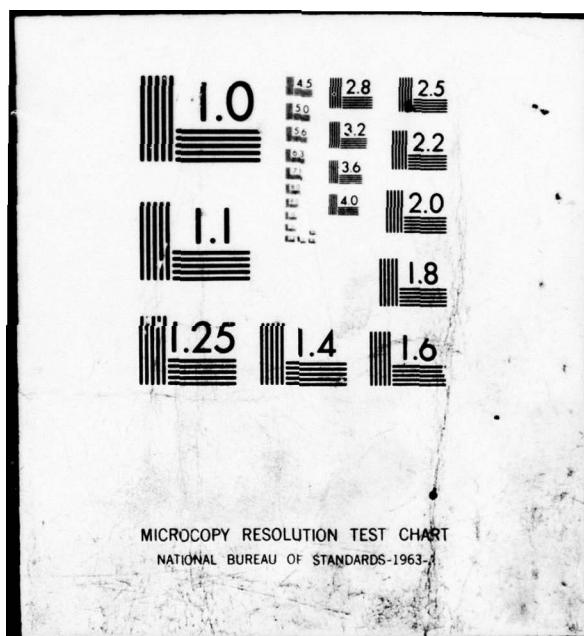
1 OF 1
AD
AO 66 5/2



END
DATE
FILMED

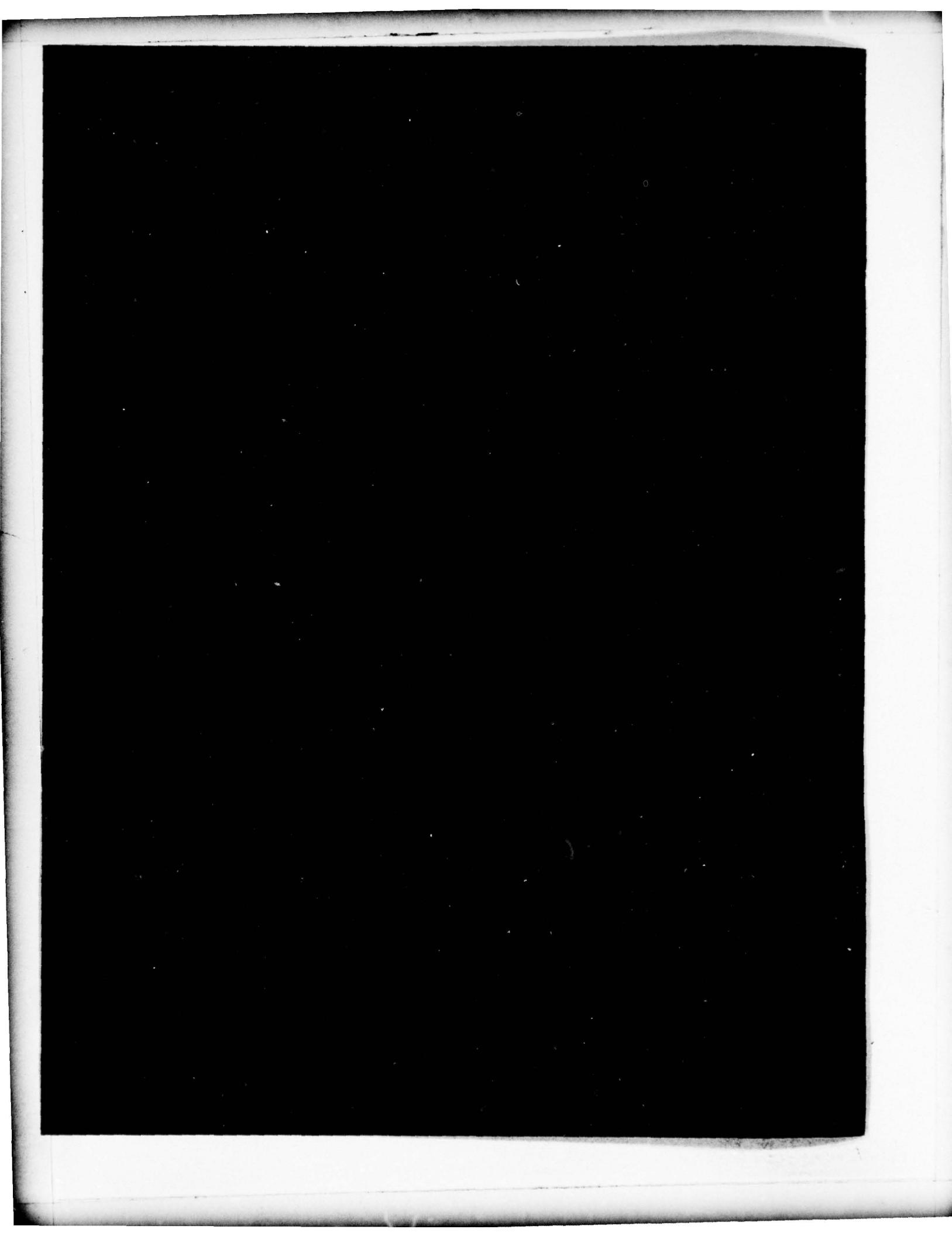
FILMED

10



DDC FILE COPY

AD A0 66512



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 14 AMSAA-TR- Technical Report No. 246	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Computer Program for Double Sweep Optimal Smoothing.	5. TYPE OF REPORT & PERIOD COVERED Technical	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Robert A. Scheder	8. CONTRACT OR GRANT NUMBER	
9. PERFORMING ORGANIZATION NAME AND ADDRESS US Army Materiel Systems Analysis Activity Aberdeen Proving Ground, MD 21005	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS DA Project No. IR765706M541	
11. CONTROLLING OFFICE NAME AND ADDRESS Commander US Army Materiel Development and Readiness Command 5001 Eisenhower Avenue Alexandria, VA 22333	12. REPORT DATE Jan 28 1979	13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) 12 74 P.	15. SECURITY CLASS. (of this report) UNCLASSIFIED	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.	17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) D D C PAPILLIP MAR 28 1979 A	
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Filter Evaluation Optimal Estimation Top Down Structured Programming	Quadratic Programming Problem Double Sweep Smoothing FORTRAN	
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report documents the double sweep smoothing program used at AMSAA to determine an aircraft's position, velocity, and acceleration from radar tracking data. It includes a tutorial guide to the underlying mathematics. A quadratic programming problem with linear constraints is formulated, reduced to a system of difference equations with initial and endpoint boundary conditions, and solved using Bryson's double sweep method. The iterative solution is programmed in simple, machine independent FORTRAN with top down structured programming. A listing is supplied and card decks are available.		

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

403 920

CONTENTS

	Page
1. INTRODUCTION	5
1.1 Purpose	5
1.2 Background.	5
1.3 Approach.	5
2. THEORY	7
2.1 Principal Definitions	7
2.2 Quadratic Programming Problem (with Linear Constraints).	10
2.3 The Solution.	11
2.3a Determination of the Final State \hat{x}_n - The Kalman Filter.	13
2.3b Backward Correcting Sweep	18
2.4 Complete Double Sweep Equations	20
3. COMPUTATION OF INPUT VARIABLES	25
3.1 Initialization Inputs \bar{P}_0 , \bar{X}_0	25
3.2 Measurement Inputs R, H, Z, and \bar{v}	26
3.3 Target Dynamical Inputs Φ , Q, R, and \bar{W}	27
3.4 Remarks on Nonlinearities	30
4. USERS' GUIDE	33
4.1 Overall Program Structure	33
4.2 User Defined Inputs	34
4.3 Identification of Program Names	35
REFERENCES	37
APPENDIX A - PROGRAM LISTING	39
DISTRIBUTION LIST.	75

ACCESSION NO.	
NTIS	White Section <input checked="" type="checkbox"/>
DOC	Buff Section <input type="checkbox"/>
UNARMED	<input type="checkbox"/>
JUSTIFICATION.....	
BY.....	
DISTRIBUTION/AVAILABILITY CODES	
B&W	AVAIL. AND/IN SPECIAL
A	

Next page is blank

79 03 27 002

A COMPUTER PROGRAM FOR DOUBLE SWEEP OPTIMAL SMOOTHING

1. INTRODUCTION

1.1 Purpose.

The purpose of this paper is to explain and document the computer routines developed by the author and used by the Army Materiel Systems Analysis Activity (AMSAA) for the optimal smoothing of aircraft tracking data.

1.2 Background.

AMSAA is the independent evaluator for the Division Air Defense Gun (DIVAD) program. One of the principal features of this program is the use of modern fire control techniques. These include optimal, Kalman filtering, routines to estimate aircraft (target) position, velocity, and acceleration from aircraft tracking data. One method used to study modern fire control consists of running target flight paths obtained during field tests through a digital simulation of the fire control and then comparing the estimates of the target state; that is, target position, velocity, and acceleration, with the corresponding "true" target state values (Reference 1). If this comparison is to be meaningful, the method used to obtain the "true" target state time history needs to provide greater accuracy than is possible to obtain through filtering alone. The method described below guarantees this needed accuracy.

1.3 Approach.

Field test tracking data consisting of target position as a function of time is processed in two steps. It is first filtered using a Kalman filter which is at least as sophisticated as the filter to be evaluated (e.g., nonlinear, adaptive, etc.). The target state estimates from the filter are then optimally corrected by sweeping in reverse chronological order through the tracking data. The corrected estimates are necessarily superior to the filtered estimates since at each instant of time, the corrected estimates were computed using all the tracking data while the filtered estimates were derived using only tracking data from earlier in time.

The general theory of optimal smoothing as considered here is fairly well established (References 2 and 3). The details, however, are often outside the background of analysts who could use (and have used) this program in other Army studies. Section 2 provides a brief but complete

tutorial of the underlying mathematics of the smoothing program which will enable the interested reader to understand and modify the program to fill his own particular need.

2. THEORY

2.1 Principal Definitions.

- All vectors and matrices are expressed in cartesian coordinates.
- Superscript -1 , $(^{-1})$, denotes matrix inverse.
- Superscript bar, $(\bar{ })$, denotes expected value.
- Superscript T, $(^T)$ denotes transpose.
- Subscript i , $(_i)$, denotes time step.
- ∇ , the $3n+1$ dimensional gradient operator, is defined to be

$$\left(\frac{\partial}{\partial x_0}, \frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_n}, \frac{\partial}{\partial v_0}, \dots, \frac{\partial}{\partial v_{n-1}}, \frac{\partial}{\partial w_1}, \dots, \frac{\partial}{\partial w_n} \right); \text{ where}$$

n is the maximum number of time steps or the maximum value of subscript i (i.e., i goes from 0 to n).

The basic problem is to determine from the tracking data the position, velocity, and acceleration or state vector of the target as a function of time in a manner which is consistent with our knowledge of aircraft dynamics and measurement uncertainty. As will be shown in Subsections 2.2 and 2.3, the process of determining estimates of the target state requires that estimates of several other vectors be simultaneously determined. These vectors are termed "derived" vectors and are defined in Table 2-1.

TABLE 2-1 DERIVED VECTORS

X : The 9 dimensional (9-D) state vector consisting of target position, velocity, and acceleration. There are $n + 1$ x's, x_i , $i = 0 \dots n$.

W : 9-D state transition error vector which can also be interpreted as the state control vector. There are n W's, w_i , $i = 0 \dots n-1$.

TABLE 2-1 DERIVED VECTORS (Continued)

V : 3-D measurement error vector. There are n V 's,
 V_i , $i = 1 \dots n$.

λ : 9-D, Lagrange Multiplier vector. There are
 $n + 1$ λ 's, λ_i , $i = 0 \dots n$.

μ : 3-D, Lagrange Multiplier vector. There are
 n μ 's, μ_i , $i = 1 \dots n$.

In addition to the "derived" vectors, we will need to define several "input" vectors and matrices. These quantities are mathematic expressions of target dynamics and measurement error. The analysis in the next 3 subsections assumes that the "input" vectors and matrices are known for all time steps, $i = 0 \dots n$. Section 3 will explain in detail how these quantities are constructed in the computer program.

TABLE 2-2 INPUT VECTORS AND MATRICES

\bar{W} : 9-D state transition error mean (or nominal control) vector. The expected value of W . There are n \bar{W} 's, \bar{W}_i , $i = 0 \dots n-1$.

\bar{V} : 3-D measurement error mean vector. The expected value of V . There are n \bar{V} 's, \bar{V}_i , $i = 1 \dots n$.

Z : 3-D target position measurement vector. The Z data as a function of time, obtained from a tracking radar during field tests, is stored in digital format on a computer tape. There are n Z 's, Z_i , $i = 1 \dots n$.

\bar{X}_o : 9-D apriori expected value of X_o .

Φ : $9 \times 9-D^2$, nonsingular, target state transition matrices. Φ_i takes the state vector from time i to time $i + 1$. There are n Φ 's, Φ_i , $i = 0 \dots n-1$. Physically, the Φ_i embody basic aircraft dynamics.

Q : $9 \times 9-D^2$ state transition error (or state control) covariance matrix. The Q_i are positive definite;

TABLE 2-2 INPUT VECTORS AND MATRICES (Continued)

$Q_i = (W_i - \bar{W}_i)(W_i - \bar{W}_i)^T$, $i = 0 \dots n-1$. Physically, Q_i measures the amount that the expected value of the state transition error vector, \bar{W}_i , can vary from the true state transition error, W_i .

R : $3 \times 3 - D^2$ measurement error covariance matrix. The R_i are positive definite; $R_i = (V_i - \bar{V}_i)(V_i - \bar{V}_i)^T$, $i = 1 \dots n$. Physically, R_i measures the amount that the expected value of the measurement error, \bar{V}_i , can vary from the true measurement error, V_i .

H : $3 \times 9 - D^2$ state to measurement transition matrix. There are n H 's, H_i , $i = 1 \dots n$. Physically, the H_i matrix takes the state vector, X_i , into the coordinate frame of the measurements, Z_i .

Γ : $9 \times 9 - D^2$ state transition error to state transition matrix. There are n Γ 's, Γ_i , $i = 0 \dots n-1$. Physically, the Γ_i matrix takes the state transition error vector, W_i , into the coordinate frame of the state vector X_i .

\bar{P}_o : $9 \times 9 - D^2$ initial state error covariance matrix. \bar{P}_o is positive definite; $\bar{P}_o = (X - \bar{X}_o)(X - \bar{X}_o)^T$. Physically, \bar{P}_o measures how much the expected value of the initial state vector, \bar{X}_o , can vary from the true initial state, X_o .

The above definitions are tied together by the dynamical target state equation (1) and by the measurement equation (2).

$$x_{i+1} = \Phi_i x_i + \Gamma_i w_i \quad i=0 \dots n-1 \quad (1)$$

$$z_i = H_i x_i + v_i \quad i=1 \dots n \quad (2)$$

From the definitions of Q and R we expect w_i to be close to \bar{w}_i when Q_i is small and similarly v_i to be close to \bar{v}_i when R_i is small. In other words, we seek estimates of x_i which satisfy equations (1) and (2) and for which the corresponding estimates of w_i and v_i are consistent with the size of Q_i and R_i . The following formulation satisfies these requirements and has the added advantage of being mathematically solvable.

2.2 Quadratic Programming Problems (with Linear Constraints).

Let

$$L = 1/2(x_o - \bar{x}_o)^T P_o^{-1} (x - x_o) + \\ 1/2 \sum_{i=0}^{n-1} [(w_i - \bar{w}_i)^T Q_i^{-1} (w_i - \bar{w}_i) + (v_{i+1} - \bar{v}_{i+1})^T R_i^{-1} (v_{i+1} - \bar{v}_{i+1})]$$

find the estimates \hat{x}_i , \hat{w}_i , \hat{v}_i of x_i , w_i , v_i respectively which minimize L subject to the constraint equations

$$\begin{aligned} \hat{x}_{i+1} &= \Phi_i \hat{x}_i + \Gamma_i \hat{w}_i \\ z_i &= H_i \hat{x}_i + \hat{v}_i \end{aligned} \quad (3)$$

The Theory of Optimal Control and Mathematical Programming (Reference 3) proves that this type of problem has a solution whenever it has a feasible solution, that is, whenever there exists \hat{x}_i , \hat{w}_i , \hat{v}_i which satisfy the constraint equations. $\hat{x}_i = 0$, $i = 0 \dots n$, $\hat{w}_i = 0$, $i = 0 \dots n$, and $\hat{v}_i = z_i$, $i = 1 \dots n$ is a feasible solution. Hence a solution to formulation (3) exists. Reference 3 further proves that there necessarily exists vectors λ_i and μ_i such that if we let

$$J = L + \sum_{i=1}^n [\lambda_{i-1}^T (\Phi_{i-1} x_{i-1} + \Gamma_{i-1} w_{i-1} - x_i) + \mu_i^T (H_i x_i + v_i - z_i)] \quad (4)$$

Then

$$\nabla J = 0. \quad (5)$$

J is called the Hamiltonian function and λ_i, μ_i are the traditional Lagrange multipliers (Reference 2 supplies proof). Since L is a convex function, equation (5) is also a sufficient condition for a feasible solution to be optimal.

2.3 The Solution.

We can highlight a useful property of J by defining the Hamiltonian sequence $J_i, i=0 \dots n$.

$$\begin{aligned} J_0 &= 1/2(x_0 - \bar{x}_0)^T P_0^{-1} (x_0 - \bar{x}_0) + 1/2(w_0 - \bar{w}_0)^T Q_0^{-1} (w_0 - \bar{w}_0) \\ &\quad + \lambda_0^T (\Phi_0 x_0 + \Gamma_0 w_0) \\ J_i &= 1/2(w_i - \bar{w}_i)^T Q_i^{-1} (w_i - \bar{w}_i) + 1/2(v_i - \bar{v}_i)^T R_i^{-1} (v_i - \bar{v}_i) \\ &\quad + \lambda_i^T (\Phi_i x_i + \Gamma_i w_i) + \mu_i^T (H_i x_i + v_i) \quad (i=1 \dots n-1) \\ J_n &= 1/2(v_n - \bar{v}_n)^T R_n^{-1} (v_n - \bar{v}_n) + \mu_n^T (H_n x_n + v_n) \end{aligned} \quad (6)$$

We find that

$$J = J_0 + \sum_{i=1}^n [J_i - \lambda_{i-1}^T x_i - \mu_i^T z_i] \quad (7)$$

$$\lambda_n = 0. \quad (8)$$

$\nabla J = 0$ implies that

$$\frac{\partial J}{\partial x_i} = 0, \frac{\partial J}{\partial w_i} = 0, \frac{\partial J}{\partial v_i} = 0$$

which in turn implies that

$$(\hat{x}_0 - \bar{x}_0)^T \bar{P}_0^{-1} + \lambda_0^T \Phi_0 = 0 \quad (9)$$

$$\lambda_i^T \Phi_i + \mu_i^T H_i - \lambda_{i-1}^T = 0 \quad i=1 \dots n-1 \quad (10)$$

$$(\hat{w}_0 - \bar{w}_0)^T Q_0^{-1} + \lambda_0^T \Gamma_0 = 0 \quad (11)$$

$$(\hat{w}_i - \bar{w}_i)^T Q_i^{-1} + \lambda_i^T \Gamma_i = 0 \quad i=1 \dots n-1 \quad (12)$$

$$(\hat{v}_i - \bar{v}_i)^T R_i^{-1} + \mu_i^T = 0 \quad i=1 \dots n \quad (13)$$

Hence we arrive at a set of simultaneous difference equations for \hat{x}_i , \hat{w}_i , and \hat{v}_i .

$$\lambda_{i-1} = H_i^T \mu_i + \Phi_i \lambda_i \quad (14)$$

$$\hat{w}_i = \bar{w}_i - Q_i \Gamma_i^T \lambda_i \quad (15)$$

$$\hat{v}_i = \bar{v}_i - R_i \mu_i \quad (16)$$

$$\hat{x}_{i+1} = \Phi_i \hat{x}_i + \Gamma_i \hat{w}_i \quad (17)$$

$$z_i = H_i \hat{x}_i + \hat{v}_i \quad (18)$$

with initial condition;

$$\hat{x}_0 = \bar{x}_0 - \bar{P}_0 \Phi_0^T \lambda_0 \quad (19)$$

and final condition $\lambda_n = 0$ (20)

These equations combine to yield our fundamental system of equations;

$$\hat{x}_{i+1} = \Phi_i \hat{x}_i - \Gamma_i Q_i \Gamma_i^T \lambda_i + \Gamma_i \bar{w}_i \quad (21)$$

$$\lambda_{i-1} = \Phi_i^T \lambda_i + H_i^T R_i^{-1} H_i \hat{x}_i - H_i^T R_i^{-1} (z_i - \bar{v}_i) \quad (22)$$

with boundary conditions

$$\hat{x}_0 = \bar{x}_0 - \bar{p}_0 \Phi_0^T \lambda_0 \quad (23)$$

$$\lambda_n = 0 \quad (24)$$

Hence the quadratic programming problem, (3), is reduced to solving equations (21) and (22) with boundary conditions (23) and (24). Note all other derived vectors in (14) through (20) can be expressed in terms of \hat{x}_i , λ_i , and input variables. For example, $\mu_i = -R_i^{-1} (z_i - H_i \hat{x}_i - \bar{v}_i)$.

The difficulty in solving equations (21) and (22) is that the boundary conditions, (23) and (24), are split -- one initial condition and one end point or final condition. Reference 4 suggests a technique for obtaining the final value, \hat{x}_n , of the optimal solution to equations (21) through (24). Having determined \hat{x}_n , we replace the split conditions (23) and (24) with the two end point conditions and can then find \hat{x}_i for all i by iterating equations (21) and (22) backwards in time. The mathematics are carried out in the next two subsections.

2.3a Determination of the Final State \hat{x}_n - The Kalman Filter. We first establish certain properties of a homogenous solution \hat{x}_i^h , and a non-homogenous particular solution, \hat{x}_i^p , whose sum is the optimal solution \hat{x}_i (i.e., $\hat{x}_i^h + \hat{x}_i^p = \hat{x}_i$ for $i = 0 \dots n$). We do not actually compute \hat{x}_i^h or \hat{x}_i^p but rather use certain of their analytical

properties to derive recursive relations which will ultimately yield the end point vector, \hat{x}_n .

For notational convenience we introduce the additional variables A_i and B_i .

$$A_i = H_i^T R_i^{-1} H_i \quad (25)$$

$$B_i = \Gamma_i Q_i \Gamma_i^T \quad (26)$$

Note the A_i and B_i are composed of input matrices and are therefore known for all i .

Consider the homogenous problem;

$$x_{i+1}^h = \Phi_i x_i^h - B_i \lambda_i^h \quad (27)$$

$$\lambda_{i-1}^h = \Phi_i^T \lambda_i^h + A_i x_i^h \quad (28)$$

with boundary conditions

$$x_0^h = -\bar{P}_0 \Phi_0^T \lambda_0^h \quad (29)$$

$$\lambda_n^h = -\lambda_n^h \quad (30)$$

The form of equations (25) through (29) suggests the relation

$$x_i^h = -P_i \Phi_i^T \lambda_i^h \quad i=0 \dots n \quad (31)$$

This is certainly true for $i = 0$. The P_i for $i > 0$ are proportionately matrices still to be determined. Substituting relation (31) into equations (27) and (28), we arrive directly at uncoupled equations for P_i , λ_i^h , and x_i^h .

$$P_{i+1} = [\Phi_i P_i \Phi_i^T + B_i] [I - A_{i+1} P_{i+1}] \quad (32)$$

$$P_0 = \bar{P}_0 \quad (33)$$

$$\lambda_{i-1}^h = [I - A_i P_i] \Phi_i^T \lambda_i^h \quad (34)$$

$$\lambda_n^h = -\lambda_n^p \quad (35)$$

$$x_i^h = -P_i \Phi_i^T \lambda_i^h \quad (36)$$

The P_i matrices play a fundamental role in our analysis.

Note that they are completely determined from initial condition (33) by iterating equation (32) forward in time.

The following definitions will be useful:

$$\tilde{P}_i = \Phi_{i-1} P_{i-1} \Phi_{i-1}^T + B_{i-1} \quad (37)$$

$$K_i = P_i H_i^T R_i^{-1} \quad (38)$$

These definitions, equations (32) and (33), and some matrix manipulation yields the following important relations;

$$P_i = [I - P_i A_i] \tilde{P}_i \quad (39)$$

$$P_i^{-1} = \tilde{P}_i^{-1} + A_i \quad (40)$$

$$P_i = \tilde{P}_i - \tilde{P}_i H_i^T [H_i \tilde{P}_i A_i^T + R_i]^{-1} H_i \tilde{P}_i \quad (41)$$

$$K_i = \tilde{P}_i H_i^T [H_i \tilde{P}_i H_i^T + R_i]^{-1} \quad (42)$$

$$P_i = (I - K_i H_i) \tilde{P}_i (I - K_i H_i)^T + K_i R_i K_i^T \quad (43)$$

We shall also find it useful to consider the particular solution to equations (21) and (22) which satisfy the initial condition

$$x_0^P = \bar{x}_0 \quad (44)$$

$$\lambda_0^P = 0 \quad (45)$$

In this case the sums

$$\hat{x}_i = x_i^h + x_i^P \quad (46)$$

$$\lambda_i = \lambda_i^h + \lambda_i^P \quad (47)$$

not only satisfy equations (21) and (22) but also the boundary conditions (23) and (24). In other words the sums defined in (46) and (47) are a representation of the solution to the quadratic programming problem.

The promised recursion equation leading to the end point vector \hat{x}_n is now within reach. To this end we define the new vectors;

$$y_i = x_i^P + p_i \Phi_i^T \lambda_i^P \quad (48)$$

$$\tilde{y}_i = \Phi_{i-1} y_{i-1} + \Gamma_{i-1} \bar{w}_{i-1} \quad (49)$$

Note that

$$y_0 = \bar{x}_0 \quad (50)$$

$$y_n = \hat{x}_n \quad (51)$$

Equation (51) assures us that finding an equation for computing y_i (by forward iteration from y_0) will lead us to the value of the end point vector \hat{x}_n . Such an iterative expression for y_i can be found by applying equations (21) and (22) to the x_i^P and λ_i^P in equation (48). This yields

$$\begin{aligned}
Y_{i+1} = & [\Phi_i - P_{i+1} A_{i+1} \Phi_i] x_i^p + \\
& [\Gamma_i - P_{i+1} A_{i+1} \Gamma_i] \bar{w}_i + \\
& K_{i+1} [z_{i+1} - \bar{v}_{i+1}] + \\
& [-B_i + P_{i+1} + P_{i+1} A_{i+1} B_i] \lambda_i^p \quad (52)
\end{aligned}$$

Using relations (37) and (39), the coefficient of the λ_i^p term in equation (52) becomes

$$[\Phi_i P_i \Phi_i^T - P_{i+1} A_{i+1} \Phi_i P_i \Phi_i^T]$$

Therefore, after recombining terms and using definition (49), we arrive at the final form of the iterative equation for the Y_i :

$$Y_i = \tilde{Y}_i + K_i [z_i - H_i \tilde{Y}_i] - K_i \bar{v}_i \quad (53)$$

$$\tilde{Y}_i = \Phi_{i-1} Y_{i-1} + \Gamma_{i-1} \bar{w}_{i-1} \quad (54)$$

with initial condition

$$Y_0 = \bar{x}_0 \quad (55)$$

the K_i are determined from equations (32), (33), and (37) through (43);

$$P_i = (I - K_i H_i) \tilde{P}_i \quad (56)$$

$$K_i = \tilde{P}_i H_i^T [H_i \tilde{P}_i H_i^T + R_i]^{-1} \quad (57)$$

$$\tilde{P}_i = \Phi_{i-1} P_{i-1} \Phi_{i-1}^T + \Gamma_{i-1} Q_i \Gamma_{i-1}^T \quad (58)$$

with initial condition

$$P_0 = \bar{P}_0 \quad (59)$$

Equations (53) through (59) taken together form a closed system completely solvable by simple forward iteration. They comprise our "forward sweep" through the data, z_i , and are traditionally called the Kalman filter. Clearly, the Y_k have the important property that they are optimal target state estimates at time $i=k$ if given only data, z_i , for $i=0\dots k$. In other words the Y_i are the best estimates that can be obtained on line (i.e., in real time). The variables Y_i and P_i will be physically interpreted further in section 2.4. For the moment we are concerned with solving equations (21) and (22) by backward iteration starting with the end point conditions

$$\hat{x}_n = y_n \quad (60)$$

$$\lambda_n = 0 \quad (61)$$

One approach is to just iterate equations (21) and (22) directly. Another, computationally more efficient approach utilizes the Y_i and P_i already computed. This latter method comprises the "Backward Sweep" and is detailed in the next subsection.

2.3b The Backward Correcting Sweep. Utilizing the preceeding analysis we can arrive directly at the optimal solution \hat{x}_i , λ_i using a backward iteration scheme where the λ_i are uncoupled from the \hat{x}_i . Using equations (22), (39), (40), (41), (42), (43), (48), and (53) we find

$$\begin{aligned} \lambda_{i-1} &= \Phi_i^T \lambda_i + A_i [x_i^h + x_i^p] - H_i^T R_i^{-1} [z_i - \bar{v}_i] \\ &= [I - A_i P_i] \Phi_i^T \lambda_i + A_i Y_i - H_i^T R_i^{-1} (z - \bar{v}_i) \\ &= [I - A_i P_i] [\Phi_i^T \lambda_i - H_i^T R_i^{-1} (z_i - \bar{v}_i - H_i Y_i)]. \end{aligned} \quad (62)$$

Using equations (21), (34), (37), (39), and (48) we find

$$\begin{aligned}
 \hat{x}_{i+1} &= \Phi_i (x_i^h + x_i^p) - B_i (\lambda_i^h + \lambda_i^p) + \Gamma_i \bar{w}_i \\
 &= y_{i+1} - P_{i+1} \Phi_{i+1}^T \lambda_{i+1}^p - \tilde{P}_{i+1} \lambda_i^h \\
 &= y_{i+1} - P_{i+1} \Phi_{i+1}^T \lambda_{i+1}
 \end{aligned}$$

or

$$\hat{x}_i = y_i - P_i \Phi_i^T \lambda_i \quad (63)$$

For completeness the end point conditions are written

$$\lambda_n = 0 \quad (64)$$

$$\hat{x}_n = y_n \quad (65)$$

Hence, the "Double Sweep Optimal Smoother" consists of sweeping through the data z_i in the forward direction using the system (53) through (59) to determine P_i and y_i $i=0\dots n$. We then sweep backwards through the z_i using equations (62) through (65) to obtain the optimal state estimates \hat{x}_i . This backward sweep can be viewed as optimally correcting the filtered estimates y_i with the data received after time step i . The estimates of the state transition error and measurement error vectors are found from equations (15) and (16)

$$\hat{w}_i = \bar{w}_i - Q_i \Gamma_i^T \lambda_i \quad (66)$$

$$\hat{v}_i = z_i - H_i \hat{x}_i \quad (67)$$

The next section derives measures of confidence for the estimates \hat{x}_i , \hat{w}_i , and \hat{v}_i .

2.4 Complete Double Sweep Equations.

If the true values of the initial state vector, the measurement error vector, and the transition error vector equalled \bar{X}_0 , \bar{V}_i , \bar{W}_i (for all i), respectively, equations (14) through (20) or (21) and (22) would imply for all i ,

$$\hat{W}_i = \bar{W}_i = \text{true transition error} \quad (68)$$

$$\hat{V}_i = \bar{V}_i = \text{true measurement error} \quad (69)$$

$$\lambda_i = 0 \quad (70)$$

$$\hat{X}_i = \text{true state vector} \quad (71)$$

$$L = 0 \quad (72)$$

In other words, the optimal estimates \hat{X} , \hat{W} , \hat{V} , necessarily take on their corresponding true values since in this case the performance criteria, L , takes on its smallest possible value, 0.

Let the operator, δ , denote the variation of a vector value from the corresponding true value. Then equations (62) through (72) imply,

$$\delta Y_i = \tilde{\delta Y}_i - K_i H_i \tilde{\delta Y}_i - K_i \delta \bar{V}_i \quad (73)$$

$$\tilde{\delta Y}_i = \Phi_{i-1} \delta Y_{i-1} + \Gamma_{i-1} \delta \bar{W}_{i-1} \quad (74)$$

$$\hat{\delta X}_i = \delta Y_i - P_i \Phi_i^T \delta \lambda_i \quad (75)$$

$$\delta \lambda_{i-1} = [I - A_i P_i] [\Phi_i \delta \lambda_i + A_i \delta Y_i + H_i^T R_i^{-1} \delta \bar{V}_i] \quad (76)$$

$$\hat{\delta W}_i = \delta \bar{W}_i - Q_i \Gamma_i^T \delta \lambda_i \quad (77)$$

$$\hat{\delta V}_i = H_i \hat{\delta X}_i \quad (78)$$

We assume that the variations of \bar{v}_i , \bar{w}_i and \bar{x}_0 from their corresponding true values are given as in the definitions of Section 2.1 and are entirely random;

$$\overline{\delta \bar{v}_i \delta \bar{v}_i^T} = R_i \delta_{ij} \quad (79)$$

$$\overline{\delta \bar{w}_i \delta \bar{w}_j^T} = Q_i \delta_{ij} \quad (80)$$

(δ_{ij} is the Kronecker delta)

$$\overline{\delta \bar{v}_i \delta \bar{w}_j^T} = 0 \quad (81)$$

$$\overline{\delta \bar{v}_i \delta \bar{x}_0^T} = 0 \quad (82)$$

$$\overline{\delta \bar{w}_i \delta \bar{x}_0^T} = 0 \quad (83)$$

$$\overline{\delta \bar{x}_0 \delta \bar{x}_0^T} = \bar{P}_0 \quad (84)$$

From equations (73) through (84) we get the important results,

$$\delta y_i = \prod_{k=0}^{i-1} c_k \delta y_0 + \sum_{k=0}^{i-1} \left(\prod_{\ell=k+1}^{i-1} c_\ell \right) c_k \quad (85)$$

where

$$c_k = p_{k+1} \tilde{p}_{k+1}^{-1} \phi_k \quad (86)$$

$$c_k = p_{k+1} \tilde{p}_{k+1}^{-1} r_k \bar{w}_k - k_{k+1} \bar{v}_{k+1} \quad (87)$$

and

$$\delta \lambda_i^T = \sum_{\ell=i+1}^n d_\ell^T \left(\prod_{k=i+1}^{\ell-1} D_k \right) \quad (88)$$

where

$$D_k = \tilde{P}_k^{-1} P_k \Phi_k^T \quad (89)$$

$$d_k = A_k Y_k + H_k^T R_k^{-1} \bar{V}_k \quad (90)$$

Using equations (85) through (90) it is straight forward, although sometimes lengthy, to show;

$$\overline{\delta Y_i \delta \bar{V}_\ell^T} = 0 \quad \ell > i \quad (91)$$

$$\overline{\delta \tilde{Y}_i \delta \bar{V}_\ell^T} = 0 \quad \ell \geq i \quad (92)$$

$$\overline{\delta Y_i \delta \bar{W}_\ell^T} = 0 \quad \ell \geq i \quad (93)$$

$$\overline{\delta \tilde{Y}_i \delta \bar{W}_\ell^T} = 0 \quad \ell \geq i \quad (94)$$

$$\overline{\delta Y_i \delta Y_\ell^T} = \overline{\delta Y_i \delta Y_i^T} \prod_{k=1}^{\ell-1} C_k^T \quad (95)$$

$$\overline{\delta Y_i \delta \lambda_i^T} - P_i \Phi_i^T \overline{\delta \lambda_i \delta \lambda_i^T} = 0 \quad (96)$$

$$\overline{(A_i \delta \lambda_i + H_i^T R_i^{-1} \delta \bar{V}_i) \delta \lambda_i^T} = 0 \quad (97)$$

$$\begin{aligned} & \overline{[A_i \delta \lambda_i + H_i^T R_i^{-1} \delta \bar{V}_i] [A_i \delta Y_i + H_i^T R_i^{-1} \delta \bar{V}_i]}^T \\ &= A_i (I - P_i A_i) \end{aligned} \quad (98)$$

$$\overline{\delta \bar{W}_\ell \delta \lambda_i^T} - Q_i \Gamma_i^T \overline{\delta \lambda \delta \lambda^T} = 0 \quad (99)$$

Using (91), (92), and (95) we get the recursive relations;

$$\overline{\delta Y_i \delta Y_i^T} = (I - K_i H_i) \overline{\delta \tilde{Y}_i \delta \tilde{Y}_i^T} (I - K_i H_i)^T + K_i R_i K_i^T \quad (100)$$

$$\overline{\delta \tilde{Y}_i \delta \tilde{Y}_i^T} = \Phi_{i-1} \overline{\delta Y_{i-1} \delta Y_{i-1}^T} \Phi_{i-1}^T + \Gamma_{i-1} Q_{i-1} \Gamma_{i-1} \quad (101)$$

$$\overline{\delta Y_0 \delta Y_0^T} = \bar{P}_0 \quad (102)$$

Comparing equations (100) through (102) with equations (33), (37), and (43) we arrive at

$$\overline{\delta Y_i \delta Y_i^T} = P_i \quad i=0 \dots n \quad (103)$$

$$\overline{\delta \tilde{Y}_i \delta \tilde{Y}_i^T} = \tilde{P}_i \quad i=1 \dots n \quad (104)$$

Hence, the P (\tilde{P}) are the covariance of the variations in Y (\tilde{Y}) due to the variations $\delta \bar{V}$, $\delta \bar{W}$. Continuing along these lines, define;

$$\Lambda_i = \overline{\delta \lambda_i \delta \lambda_i^T} \quad (105)$$

$$SP_i = \overline{\delta \hat{X}_i \delta \hat{X}_i^T} \quad (106)$$

$$SQ_i = \overline{\delta \hat{W}_i \delta \hat{W}_i^T} \quad (107)$$

$$SR_i = \overline{\delta \hat{V}_i \delta \hat{V}_i^T} \quad (108)$$

SP, SQ, SR are the covariances of the errors in the estimates \hat{X} , \hat{W} , \hat{V} , respectively. Using equations (73) through (78) and (91) through (99) we easily find;

$$SP_i = P_i - P_i \Phi_i^T \Lambda_i \Phi_i P_i \quad (109)$$

$$SQ_i = Q_i - Q_i \Gamma_i \Lambda_i \Gamma_i^T Q_i \quad (110)$$

$$SR_i = H_i S P_i H_i^T \quad (111)$$

$$\Lambda_{i-1} = (I - P_i A_i)^T \Phi_i^T \Lambda_i \Phi_i (I - P_i A_i)$$

$$+ A_i (I - P_i A_i) \quad (112)$$

$$\Lambda_n = 0 \quad (113)$$

The forward sweep equations, (53) through (59), and the complete backward sweep equations, (62) through (67) and (109) through (113), form the "Double Sweep Optimal Smoother." Note that the double sweep equations require values for the input variables \bar{X}_0 , \bar{P}_0 , \bar{W} , \bar{V} , R and Q. The next section describes how these quantities are computed.

3. COMPUTATION OF INPUT VARIABLES

In order for the iterative process of the double sweep smoother to be initialized and maintained, the \bar{X}_0 , \bar{P}_0 , \bar{W}_i , \bar{V}_i , R_i , Q_i , and Φ_i must be supplied for all i . A complete analysis of these quantities is given in "Adaptive Estimation" (Reference 5). A summary is given below.

3.1 Initialization Inputs \bar{P}_0 and \bar{X}_0 .

All nine components of \bar{X}_0 are set equal to zero. \bar{P}_0 has all off diagonal elements set to zero. Its first three diagonal elements, corresponding to initial position uncertainty are set very large. The initial velocity uncertainty is large but recognizes that the aircraft is subsonic while the initial acceleration uncertainty assumes that the aircraft is not in a severe maneuver at initial detection. Smoother (and filter) estimates are not very sensitive to small changes in \bar{P}_0 as long as the diagonal elements are large.

$$\bar{X}_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \bar{P}_0 = \begin{pmatrix} (3000)^2 I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & (240)^2 I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & (25)^2 I_{3 \times 3} \end{pmatrix}$$

The assumed values of the initial state and the corresponding blocks of \bar{P}_0 given above can easily be changed by the user.

3.2 Measurement Inputs R, H, Z, and \bar{V} .

Since most of the tracking data that we use is supplied in a fixed inertial cartesian coordinate frame, the smoother assumes that the z_i are expressed in the same frame as the state vector, although it can be easily modified to account for any measurement frame. The model of the measurement errors in the program accounts for constant variance angular errors and also constant variance linear, glint type, errors in both azimuth and elevation and for a fixed variance linear error in range. The covariance matrix of these errors is computed in the radar line of sight frame and then expressed, via a similarity transformation, in the fixed reference frame. The reference to sensor frame coordinate transformation TRS_i is computed at each i using the latest estimate, Y_i , of the target position vector during the forward sweep. The measurement related matrices are;

$$H_i = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$TRS_i = \begin{pmatrix} 1_{Y \div |Y|} & 2_{Y \div |Y|} & 3_{Y \div |Y|} \\ -2_{Y \div |Yg|} & 1_{Y \div |Yg|} & 0 \\ -1_{Y^3 Y \div |Y||Yg|} & -2_{Y^3 Y \div |Y||Yg|} & |Yg| \div |Y| \end{pmatrix}$$

(the subscript i's on the Y's are omitted for notational convenience). Where

$$|Y| = \sqrt{1_Y^2 + 2_Y^2 + 3_Y^2}$$

$$|Yg| = \sqrt{1_Y^2 + 2_Y^2}$$

and upper left superscript denotes component.

$$R_i = TRS_i^T \begin{pmatrix} \text{sigrzs}_i & 0 & 0 \\ 0 & \text{sigrys}_i & 0 \\ 0 & 0 & \text{sigrzs}_i \end{pmatrix} TRS_i$$

where

$$\text{sigrzs}_i = \text{sigmr}^2$$

$$\text{sigrys}_i = (\text{stdazl})^2 + (|Y_i| \text{stdaz2})^2$$

$$\text{sigrzs}_i = (\text{stdell})^2 + (|Y_i| \text{stdel2})^2.$$

sigmr , stdazl , stdaz2 , stdell , stdel2 , are constants set by the user. Nominal values are 5(meters), 1(meter), .0005 (mils \div 1000), 1(meter), .0005(mils \div 1000), respectively.

We assume that the tracking radar which generated the Z_i was well calibrated and therefore had zero bias. In other words,

$$\bar{V}_i = 0 \quad i=1 \dots n.$$

3.3 The Target Dynamical Inputs Φ , Q , Γ , and \bar{W} .

Φ_i represents the dynamics of an aircraft which is controlled at time step i through its acceleration rate, $\Gamma_i W_i$. Alternatively, we can look on Φ_i as representing an aircraft whose acceleration is constant over any particular transition, $i \rightarrow i+1$, where the error in this representation is given by $\Gamma_i W_i$.

$$\Phi_i = \begin{pmatrix} I_{3 \times 3} & I_{3 \times 3} \Delta t & .5 I_{3 \times 3} \Delta t^2 \\ 0_{3 \times 3} & I_{3 \times 3} & I_{3 \times 3} \Delta t \\ 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} \end{pmatrix}$$

where Δt = time step between iterations. Note that Φ is constant over all i .

The computations of the inputs \bar{W}_i and Q_i are complicated in that they are physically related to the aircraft body frame variables of thrust rate, g rate, and roll rate and their corresponding covariances. The transition from body frame dynamics to reference frame dynamics is worked out in Reference 5. At any particular instant, a circle is matched to the target's flight path with the properties that the component of target acceleration perpendicular to the velocity vector points towards the circle's center. The target's body frame has its first component axis along the velocity vector and its second component axis along the normal acceleration vector. It is also called the Frenet frame. Γ is the body, or Frenet frame, to reference frame coordinate transformation. It is computed at each i based on the latest available state estimate. Γ 's matrix components are;

$$11_{\Gamma} = \frac{4_y}{v}$$

$$21_{\Gamma} = \frac{5_y}{v}$$

$$31_{\Gamma} = \frac{6_y}{v}$$

$$12_{\Gamma} = (7_y - 4_y \Delta t) \div gn$$

$$22_{\Gamma} = (8_y - 5_y \Delta t) \div gn$$

$$32_{\Gamma} = (9_y - 6_y \Delta t) \div gn$$

$$13_{\Gamma} = 21_{\Gamma} 32_{\Gamma} - 31_{\Gamma} 22_{\Gamma}$$

$$23_{\Gamma} = 12_{\Gamma} 31_{\Gamma} - 11_{\Gamma} 32_{\Gamma}$$

$$33_{\Gamma} = 11_{\Gamma} 22_{\Gamma} - 12_{\Gamma} 21_{\Gamma}$$

where

$$v = (\frac{4_y^2 + 5_y^2 + 6_y^2}{3})^{1/2}$$

$$k = [({}^5Y^9Y - {}^6Y^8Y)^2 + ({}^4Y^9Y - {}^7Y^6Y)^2 + ({}^4Y^8Y - {}^5Y^7Y)^2]^{1/2} \div v^3$$

$$at = ({}^4Y^7Y + {}^5Y^8Y - {}^6Y^9Y) \div v^2$$

$$gn = kv^2.$$

The computation of \bar{W} is based on the assumption that the aircraft flies along the Frenet circle over time Δt .

$$\bar{W} = \begin{pmatrix} -k^2 v^3 \Delta t^3 \div 6 \\ k v v \Delta t^3 \div 6 \\ 0 \\ -k^2 v^3 \Delta t^2 \div 2 \\ k v v \Delta t^2 \div 2 \\ 0 \\ -k^2 v^3 \Delta t \\ k v v \Delta t \\ 0 \end{pmatrix}$$

where

$$\dot{v} = ({}^4Y^7Y + {}^4Y^8Y + {}^6Y^9Y) \div v.$$

The corresponding transition error covariance is

$$Q = \begin{pmatrix} QAP \Delta t^4 \div 36 & QAP \Delta t^3 \div 12 & QAP \Delta t^2 \div 2 \\ QAP \Delta t^3 \div 12 & QAP \Delta t^2 \div 2 & QAP \Delta t \\ QAP \Delta t^2 \div 2 & QAP \Delta t & QAP \end{pmatrix}$$

where

$$QAP = \begin{pmatrix} (C_1 \Delta t)^2 & 0 & 0 \\ 0 & (C_2 \Delta t)^2 & 0 \\ 0 & 0 & (gn * C_3 \Delta t)^2 \end{pmatrix}.$$

C_1, C_2, C_3 are constants set by the user corresponding to the one standard deviation (STD) of thrust rate, g rate, and roll rate, respectively. Nominal values are 1.0(meters/sec³), 7.0(meters/sec³), and 1.0(radians/sec respectively. Note, for notational convenience the i subscripts have been omitted in all the above equations.

3.4 Remarks on Nonlinearities.

The input variables can in principle be defined in a variety of ways. For example, they could each be referred to any number of possibly different coordinate systems. Unfortunately, the nature of the tracking problem will result in any, but the most simple minded, formulation having nonlinear portions. The investigator can influence where these nonlinearities occur. In our formulation they have been shifted to the coordinate transition matrices TRS and Γ .

Nonlinearities have two effects. On the forward sweep, Γ and TRS depend on the most recent state estimates, Y_i , for their computation. Therefore the P_i cannot be a priori computed but must be determined simultaneously with the Y_i . Secondly, we must change our interpretation of the δ operator in equations (73) through (108) from that of variation to that of small variation.

The nonlinearities raise some question as to the adequacy of the \bar{W} and Q computations, particularly for severely jinking targets where lags in the forward sweep estimates may become large. Conceivably we can reduce the lags by increasing Q , that is increasing C_1, C_2, C_3 , and/or running the smoother back and forth over the data several times, each time starting with \bar{X}_0 and \bar{P}_0 from the previous

backward sweep. Doubtless the reader can think of other schemes for reducing the effects of nonlinearities.

Next page is blank

4. USERS' GUIDE

The program uses Top Down Structured Programming (TDSP) and is written in simple, machine independent FORTRAN. In line with TDSP, the coding uses only five logical structures, is self-documented (requiring no flow charts) and is completely straightforward (avoiding "programming tricks") (Reference 6). A brief description of the program's overall structure and a few remarks on input data are all a perspective user will need for a guide.

4.1 Overall Structure.

A schematic diagram of the functional hierarchy is given in Figure 4.1.

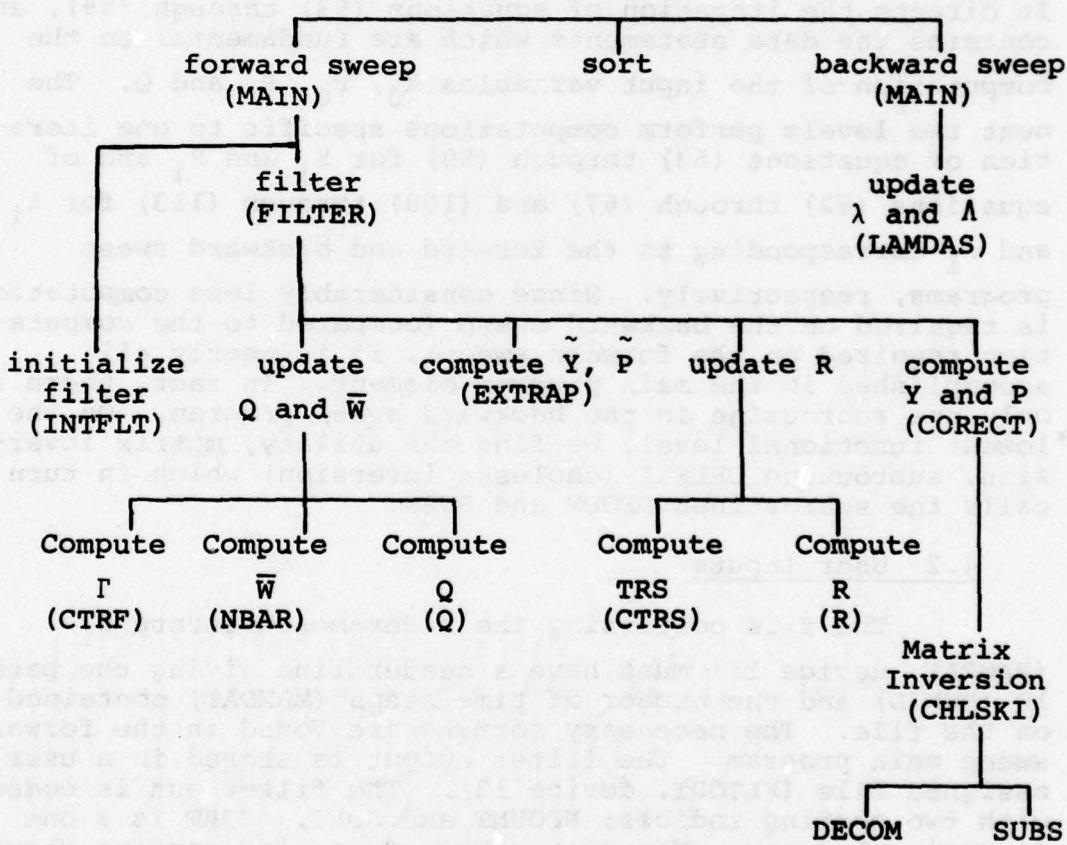


Figure 4.1 Programming Structure (Subroutine Name)

The first level consists of three independent programs, the forward sweep, sort, and backward sweep programs. The forward sweep reads the tracking data from a user selected storage device (e.g., FASTRAND, tape, disc), filters it, and writes the results on user selected devices. The sort program reverses the chronological order of the filtered output in preparation for the backward sweep. The user is urged to consult his own computer facility for a machine optimized sort routine. For example, AMSAA's Air Defense Evaluation Branch uses a sort program optimized for the Univac 1108 consisting of one executive command, @MISD* LLB\$.SORTSDF 12., 12.,#records, #chars/per, KEY/127/4.D, KEY/131/2.A. The backward sweep reads the filtered output in reverse chronological order and writes the final smoothed results on user selected devices. The user may wish to use SORT again since the smoothed results are in reverse chronological order.

The next functional level is the FILTER subroutine. It directs the iteration of equations (53) through (59), and contains the data statements which are fundamental to the computation of the input variables \bar{X}_0 , \bar{P}_0 , R, and Q. The next two levels perform computations specific to one iteration of equations (53) through (59) for Y_i and P_i and of equations (62) through (67) and (109) through (113) for λ_i and Λ_i corresponding to the forward and backward sweep programs, respectively. Since considerably less computation is required on the backward sweep (compared to the computation required on the forward sweep), it is nearly all accomplished in the main program element. In fact, there is only one subroutine in the backward sweep program. On the lowest functional level, we find the utility, matrix inversion, subroutine CHLSKI (choleski inversion) which in turn calls the subroutines DECOM and SUBS.

4.2 User Inputs.

The file containing the radar measurements z_i (RAWDAT, device 11) must have a header line giving the path ID (RUNID) and the number of time steps (MAXDAT) contained on the file. The necessary formats are found in the forward sweep main program. The filter output is stored in a user assigned file (FLTOOUT, device 12). The filter out is coded with two sorting indices; NCOUNT and SORT. SORT is a one dimensional array. The user selected sorting routine should sort in decreasing numerical order on NCOUNT and in normal ascending order on SORT. The backward sweep program reads the sorted (reverse chronological order) data from FLTOOUT and writes the smoothed data in a user assigned file (SMTOUT, device 13).

The data time step, DELTAT, must be set in each of the two main program elements. Subroutine INTFLT contains all the remaining variables that are to be set by the user. They are;

CONE, the STD of target thrust rate,
CTWO, the STD of target g rate,
CTHREE, the STD of target roll rate,
SIGMR, the STD of range measurement error,
STDAZ1, the STD of linear azimuth measurement error,
STDAZ2, the STD of angular azimuth measurement error,
STDELL, the STD of linear elevation measurement error,
STDEL2, the STD of angular elevation measurement error,
PVAL(i), i = 1...9, are the STDs of the initial state vector in the X, Y, Z directions of position, velocity, and acceleration, respectively.

4.3 Identification of Program Names.

This section identifies the FORTRAN names of the principal variables used in the program with the corresponding symbols used in the theory of sections 2 and 3.

<u>Program Name</u>	<u>Theory Name</u>
BARN	\bar{W}
CAPLAM	Λ
DELTAT	Δt
HKAL	H
LAMDA	λ
PKAL	\tilde{P}, P
QA	$\Gamma QAP \Gamma^T$
QAP	QAP
QKAL	$\Gamma Q \Gamma^T$

<u>Program Name</u>	<u>Theory Name</u>
RESUL	Z-HY
RMEAS	R
RMEASI	R⁻¹
SAB	A
SMTBRN	Γ \hat{W}
SMTP	SP
SMTQ	Γ SO Γ^T
SMTSPS	\hat{X}
SPS	\tilde{Y}, \tilde{Y}
TRS	TRS
TRV	Γ^T
ZMEAS	Z

<u>Program Name</u>	<u>Theory Name</u>
RESUL	RMEAS
RMEAS	RMEAS
SAB	RATRN
SMTBRN	RATRN
SMTP	RATRN
SMTQ	RATRN
SMTSPS	RATRN
SPS	RATRN
TRS	RATRN
TRV	RATRN
ZMEAS	RATRN

REFERENCES

1. Meredith, J., Schéder, R., Lufkin, B., Evaluation of the Gun Low Altitude Fire Control Test Bed (GLAADS), AMSAA Technical Report No. 149, April 1977.
2. Bryson, A.E., Jr., and Yu-Chi Ho, Applied Optimal Control, Waltham, MA, Ginn and Company, 1969.
3. Canon, M.D., Clifton, D.C., Polak, E., Theory of Optimal Control and Mathematical Programming, New York, NY, McGraw-Hill, 1970.
4. Bryson, A.E., and Frazier, M., Smoothing for Linear and Nonlinear Dynamic Systems, Proceedings of the Optimum System Synthesis Conference, Wright-Patterson Air Force Base, Ohio, September 1963.
5. Scheder, R.A., Adaptive Estimation, AMSAA Technical Report No. 166, December 1976.
6. Mills, H.D., Mathematical Foundations for Structured Programming, IBM Corporation, Federal Systems Division, Gaithersburg, MD, 1972.

Next page is blank

APPENDIX A

PROGRAM LISTING

INDEX

<u>Table</u>	<u>Title</u>	<u>Page</u>
A-1	Forward Sweep Main Program	41
A-2	Backward Sweep Main Program	44
A-3	Subroutine CHLSKI	49
A-4	Subroutine CORECT	50
A-5	Subroutine CTRF	52
A-6	Subroutine CTRS	55
A-7	Subroutine DECOM	57
A-8	Subroutine EXTRAP	58
A-9	Subroutine FILTER	60
A-10	Subroutine INTFLT	61
A-11	Subroutine LAMDAS	64
A-12	Subroutine NBAR	67
A-13	Subroutine Q	69
A-14	Subroutine R	71
A-15	Subroutine SUBS	73

Table A-1 Forward Sweep Main Program

```

1 C THIS PROGRAM ESTIMATES TARGET POSITION, VELOCITY, AND ACCELERATION
2 C BY SWEEPING THRU RAW TRACKING (POSITION) DATA USING A SELF
3 C ADAPTING, NON-LINEAR, KALMAN FILTER.
4 C
5 C REFERENCES
6 C
7 C 1. SCHEDER R.A., ADAPTIVE ESTIMATION,
8 C AAMSAA TECHNICAL REPORT 166
9 C APG, MD 1975.
10 C
11 C 2. BRYSON, A.E. JR., AND HO YU-CHI,
12 C APPLIED OPTIMAL CONTROL, GINN AND COMPANY,
13 C WALTHAM, MA 1969, CHAPTER 12
14 C
15 C COMMON /KAL10 / RESUL,BARN ,SPS 'PKAL ,QA
16 C ,ZMEAS ,HKAL ,PHI ,RMEAS ,RMEASI ,QKAL
17 C
18 C COMMON /KALDAT/ PVAL ,DELTAT,CONE ,CTWO ,CTHREE ,SIGMR
19 C ,STD1,STD2,STD1,STD2
20 C
21 C REAL RESUL(3)
22 C A BARN(9)
23 C B SPS(9)
24 C C PKAL(9,9)
25 C D QA(3,3)
26 C E RMEAS(3,3)
27 C F QKAL(9,9)
28 C G ZMEAS(3)
29 C H HKAL(3,9)
30 C I PHI(9,9)
31 C J RMEAS(3,3)
32 C
33 C REAL PVAL(9)
34 C A DELTAT
35 C B CONE
36 C C CTHREE
37 C D CTWOC
38 C E SIGMR
39 C F STD1
40 C G STD2
41 C H STD1
42 C I STD2
43 C
44 C REAL ADDT(3)
45 C
46 C INTEGER RAWDAT
47 C A FLTOUT
48 C B RUNID
49 C C MAXDAT
50 C D SORT(20)
51 C
52 C 100 FORMAT (4F10.2)
53 C 200 FORMAT (110,F10.2,6F14.7,22X,14.12,/(9F14.7,14,12))
54 C 300 FORMAT (1X,14,F10.2,9F9.1,2X,319.2)
55 C 400 FORMAT (1X,'STATE ERROR COVARIANCE',9(/,1X,9F10.2,
56 A ,/1X,'ACCELERATION TRANSITION ERROR COVARIANCE'

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```
B      C      .3(/,1X,3F8.2)
58      500      C      FORMAT  ./,1X, MEASUREMENT ERROR COVARIANCE',3(/,1X,3F8.2)
59      600      C      FORMAT  (110,F10.2,6F12.3,./,9F12.3,./,9F12.3)
60      600      C      FORMAT  (14,1X,14)
61      700      C      FORMAT  (110,116X,14,12)
62      C
63      C      INITIALIZATION
64      C
65      C      DELTAT = 0.05
66      C      RANDAT = 11
67      C      FLTOUT = 12
68      C
69      C      REWIND RANDAT
70      C      REWIND FLTOUT
71      C
72      C      IO = 0
73      C      NCOUNT = 0
74      C
75      1000      C      DO 1000 I = 1,20
76      C      SORT(I) = 1
77      C      CONTINUE
78      C
79      C      READ  (RANDAT,600)  RUNDAT
80      C      WRITE (FLTOUT,700)  RUNDAT, MAXDAT, IO
81      C      WRITE (6,600)  RUNDAT, MAXDAT
82      C
83      C      CALL INTFLT
84      C
85      C      FORWARD SWEEP THRU RAW DATA
86      C
87      3000      C      DO 3000 II = 1, MAXDAT
88      C      READ  (RANDAT,100)  TIME  , (ZMEAS(I),I=1,3)
89      C      NCOUNT = NCOUNT + 1
90      C      CALL FILTER
91      C
92      C      OUTPUT
93      C
94      C      WRITE(FLTOUT,200)  NCOUNT, TIME, (RESUL(I),I=1,3)
95      A      C      ,(BARN(),I=7,9),NCOUNT, SORT(1)
96      B      C      ,(SPS(1),I=1,9), NCOUNT, SORT(2)
97      C      ,((PKAL(I,J),J=1,9),NCOUNT, SORT(1+2),I=1,9)
98      D      C      ,((QA(I,J),J=1,3),I=1,3), NCOUNT, SORT(12)
99      E      C      ,((RMEAS(I,J),J=1,3),I=1,3), NCOUNT, SORT(13)
100      C
101      C      ADDT(1) = BARN(7) / DELTAT
102      C      ADDT(2) = BARN(8) / DELTAT
103      C      ADDT(3) = BARN(9) / DELTAT
104      C
105      C      WRITE (6,300)  NCOUNT, TIME, (SPS(I),I=1,9), (ADDT(I),I=1,3)
106      C
107      C      IF  ( MOD((NCOUNT - 1),20) .NE. 0 )  GO TO 2500
108      A      C      WRITE (6,400)  ((PKAL(I,J),J=1,9),I =1,9)
109      B      C      ,((QKAL(I,J),J=7,9),I =7,9)
110      C      ,((RMEAS(I,J),J=1,3),I =1,3)
111      2500      C      CONTINUE
112      3000      C      CONTINUE
113      C
```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

114 C
115 ENDFILE FLTOUT
116 STOP
117 END

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

Table A-2 Backward Sweep Main Program

1 THIS PROGRAM SMOOTH'S TRACKING DATA BY SWEEPING BACKWARDS
 2 C THROUGH PREVIOUSLY FILTERED DATA.
 3 C
 4 C REFERENCES
 5 C
 6 C . MEREDITH J., SCHEDER R., LUFKIN B., EVALUATION OF
 7 C THE GUN LOW ALTITUDE AIR DEFENSE FIRE CONTROL TEST
 8 C BED (GLAAD), AMSAA TECHNICAL REPORT 149,
 9 C APG, MD 1977. CHAPTER 3
 10 C
 11 C 2. BRYSON, A.E., JR., AND HO, YU-CHI,
 12 C APPLIED OPTIMAL CONTROL, GINN AND COMPANY,
 13 C WALTHAM, MA 1969. CHAPTER 13
 14 C
 15 C COMMON /KALIO / RESDUL,BARN ,SPS ,PKAL ,QA ,RMEAS ,QKAL
 16 C A COMMON /ABIO / LAMDA ,CAPLAM
 17 C
 18 C DOUBLE PRECISION LAMDA(9)
 19 C
 20 C A DOUBLE PRECISION RESDUL(3)
 21 C A BARN(9)
 22 C B SPS(9)
 23 C C PKAL(9,9)
 24 C D QA(3,3)
 25 C E RMEAS(3,3)
 26 C F QKAL(9,9)
 27 C G ZMEAS(3)
 28 C H HKAL(3,9)
 29 C I PHI(9,9)
 30 C J RMEAS(3,3)
 31 C
 32 C
 33 C
 34 C
 35 C
 36 C
 37 C
 38 C
 39 C
 40 C
 41 C
 42 C
 43 C
 44 C
 45 C
 46 C
 47 C
 48 C
 49 C
 50 C
 51 C
 52 C
 53 C
 54 C
 55 C
 56 C
 57 C
 58 C
 59 C
 60 C
 61 C
 62 C
 63 C
 64 C
 65 C
 66 C
 67 C
 68 C
 69 C
 70 C
 71 C
 72 C
 73 C
 74 C
 75 C
 76 C
 77 C
 78 C
 79 C
 80 C
 81 C
 82 C
 83 C
 84 C
 85 C
 86 C
 87 C
 88 C
 89 C
 90 C
 91 C
 92 C
 93 C
 94 C
 95 C
 96 C
 97 C
 98 C
 99 C
 100 C
 101 C
 102 C
 103 C
 104 C
 105 C
 106 C
 107 C
 108 C
 109 C
 110 C
 111 C
 112 C
 113 C
 114 C
 115 C
 116 C
 117 C
 118 C
 119 C
 120 C
 121 C
 122 C
 123 C
 124 C
 125 C
 126 C
 127 C
 128 C
 129 C
 130 C
 131 C
 132 C
 133 C
 134 C
 135 C
 136 C
 137 C
 138 C
 139 C
 140 C
 141 C
 142 C
 143 C
 144 C
 145 C
 146 C
 147 C
 148 C
 149 C
 150 C
 151 C
 152 C
 153 C
 154 C
 155 C
 156 C
 157 C
 158 C
 159 C
 160 C
 161 C
 162 C
 163 C
 164 C
 165 C
 166 C
 167 C
 168 C
 169 C
 170 C
 171 C
 172 C
 173 C
 174 C
 175 C
 176 C
 177 C
 178 C
 179 C
 180 C
 181 C
 182 C
 183 C
 184 C
 185 C
 186 C
 187 C
 188 C
 189 C
 190 C
 191 C
 192 C
 193 C
 194 C
 195 C
 196 C
 197 C
 198 C
 199 C
 200 C
 201 C
 202 C
 203 C
 204 C
 205 C
 206 C
 207 C
 208 C
 209 C
 210 C
 211 C
 212 C
 213 C
 214 C
 215 C
 216 C
 217 C
 218 C
 219 C
 220 C
 221 C
 222 C
 223 C
 224 C
 225 C
 226 C
 227 C
 228 C
 229 C
 230 C
 231 C
 232 C
 233 C
 234 C
 235 C
 236 C
 237 C
 238 C
 239 C
 240 C
 241 C
 242 C
 243 C
 244 C
 245 C
 246 C
 247 C
 248 C
 249 C
 250 C
 251 C
 252 C
 253 C
 254 C
 255 C
 256 C
 257 C
 258 C
 259 C
 260 C
 261 C
 262 C
 263 C
 264 C
 265 C
 266 C
 267 C
 268 C
 269 C
 270 C
 271 C
 272 C
 273 C
 274 C
 275 C
 276 C
 277 C
 278 C
 279 C
 280 C
 281 C
 282 C
 283 C
 284 C
 285 C
 286 C
 287 C
 288 C
 289 C
 290 C
 291 C
 292 C
 293 C
 294 C
 295 C
 296 C
 297 C
 298 C
 299 C
 300 C
 301 C
 302 C
 303 C
 304 C
 305 C
 306 C
 307 C
 308 C
 309 C
 310 C
 311 C
 312 C
 313 C
 314 C
 315 C
 316 C
 317 C
 318 C
 319 C
 320 C
 321 C
 322 C
 323 C
 324 C
 325 C
 326 C
 327 C
 328 C
 329 C
 330 C
 331 C
 332 C
 333 C
 334 C
 335 C
 336 C
 337 C
 338 C
 339 C
 340 C
 341 C
 342 C
 343 C
 344 C
 345 C
 346 C
 347 C
 348 C
 349 C
 350 C
 351 C
 352 C
 353 C
 354 C
 355 C
 356 C
 357 C
 358 C
 359 C
 360 C
 361 C
 362 C
 363 C
 364 C
 365 C
 366 C
 367 C
 368 C
 369 C
 370 C
 371 C
 372 C
 373 C
 374 C
 375 C
 376 C
 377 C
 378 C
 379 C
 380 C
 381 C
 382 C
 383 C
 384 C
 385 C
 386 C
 387 C
 388 C
 389 C
 390 C
 391 C
 392 C
 393 C
 394 C
 395 C
 396 C
 397 C
 398 C
 399 C
 400 C
 401 C
 402 C
 403 C
 404 C
 405 C
 406 C
 407 C
 408 C
 409 C
 410 C
 411 C
 412 C
 413 C
 414 C
 415 C
 416 C
 417 C
 418 C
 419 C
 420 C
 421 C
 422 C
 423 C
 424 C
 425 C
 426 C
 427 C
 428 C
 429 C
 430 C
 431 C
 432 C
 433 C
 434 C
 435 C
 436 C
 437 C
 438 C
 439 C
 440 C
 441 C
 442 C
 443 C
 444 C
 445 C
 446 C
 447 C
 448 C
 449 C
 450 C
 451 C
 452 C
 453 C
 454 C
 455 C
 456 C
 457 C
 458 C
 459 C
 460 C
 461 C
 462 C
 463 C
 464 C
 465 C
 466 C
 467 C
 468 C
 469 C
 470 C
 471 C
 472 C
 473 C
 474 C
 475 C
 476 C
 477 C
 478 C
 479 C
 480 C
 481 C
 482 C
 483 C
 484 C
 485 C
 486 C
 487 C
 488 C
 489 C
 490 C
 491 C
 492 C
 493 C
 494 C
 495 C
 496 C
 497 C
 498 C
 499 C
 500 C
 501 C
 502 C
 503 C
 504 C
 505 C
 506 C
 507 C
 508 C
 509 C
 510 C
 511 C
 512 C
 513 C
 514 C
 515 C
 516 C
 517 C
 518 C
 519 C
 520 C
 521 C
 522 C
 523 C
 524 C
 525 C
 526 C
 527 C
 528 C
 529 C
 530 C
 531 C
 532 C
 533 C
 534 C
 535 C
 536 C
 537 C
 538 C
 539 C
 540 C
 541 C
 542 C
 543 C
 544 C
 545 C
 546 C
 547 C
 548 C
 549 C
 550 C
 551 C
 552 C
 553 C
 554 C
 555 C
 556 C
 557 C
 558 C
 559 C
 560 C
 561 C
 562 C
 563 C
 564 C
 565 C
 566 C
 567 C
 568 C
 569 C
 570 C
 571 C
 572 C
 573 C
 574 C
 575 C
 576 C
 577 C
 578 C
 579 C
 580 C
 581 C
 582 C
 583 C
 584 C
 585 C
 586 C
 587 C
 588 C
 589 C
 590 C
 591 C
 592 C
 593 C
 594 C
 595 C
 596 C
 597 C
 598 C
 599 C
 600 C
 601 C
 602 C
 603 C
 604 C
 605 C
 606 C
 607 C
 608 C
 609 C
 610 C
 611 C
 612 C
 613 C
 614 C
 615 C
 616 C
 617 C
 618 C
 619 C
 620 C
 621 C
 622 C
 623 C
 624 C
 625 C
 626 C
 627 C
 628 C
 629 C
 630 C
 631 C
 632 C
 633 C
 634 C
 635 C
 636 C
 637 C
 638 C
 639 C
 640 C
 641 C
 642 C
 643 C
 644 C
 645 C
 646 C
 647 C
 648 C
 649 C
 650 C
 651 C
 652 C
 653 C
 654 C
 655 C
 656 C
 657 C
 658 C
 659 C
 660 C
 661 C
 662 C
 663 C
 664 C
 665 C
 666 C
 667 C
 668 C
 669 C
 670 C
 671 C
 672 C
 673 C
 674 C
 675 C
 676 C
 677 C
 678 C
 679 C
 680 C
 681 C
 682 C
 683 C
 684 C
 685 C
 686 C
 687 C
 688 C
 689 C
 690 C
 691 C
 692 C
 693 C
 694 C
 695 C
 696 C
 697 C
 698 C
 699 C
 700 C
 701 C
 702 C
 703 C
 704 C
 705 C
 706 C
 707 C
 708 C
 709 C
 710 C
 711 C
 712 C
 713 C
 714 C
 715 C
 716 C
 717 C
 718 C
 719 C
 720 C
 721 C
 722 C
 723 C
 724 C
 725 C
 726 C
 727 C
 728 C
 729 C
 730 C
 731 C
 732 C
 733 C
 734 C
 735 C
 736 C
 737 C
 738 C
 739 C
 740 C
 741 C
 742 C
 743 C
 744 C
 745 C
 746 C
 747 C
 748 C
 749 C
 750 C
 751 C
 752 C
 753 C
 754 C
 755 C
 756 C
 757 C
 758 C
 759 C
 760 C
 761 C
 762 C
 763 C
 764 C
 765 C
 766 C
 767 C
 768 C
 769 C
 770 C
 771 C
 772 C
 773 C
 774 C
 775 C
 776 C
 777 C
 778 C
 779 C
 780 C
 781 C
 782 C
 783 C
 784 C
 785 C
 786 C
 787 C
 788 C
 789 C
 790 C
 791 C
 792 C
 793 C
 794 C
 795 C
 796 C
 797 C
 798 C
 799 C
 800 C
 801 C
 802 C
 803 C
 804 C
 805 C
 806 C
 807 C
 808 C
 809 C
 810 C
 811 C
 812 C
 813 C
 814 C
 815 C
 816 C
 817 C
 818 C
 819 C
 820 C
 821 C
 822 C
 823 C
 824 C
 825 C
 826 C
 827 C
 828 C
 829 C
 830 C
 831 C
 832 C
 833 C
 834 C
 835 C
 836 C
 837 C
 838 C
 839 C
 840 C
 841 C
 842 C
 843 C
 844 C
 845 C
 846 C
 847 C
 848 C
 849 C
 850 C
 851 C
 852 C
 853 C
 854 C
 855 C
 856 C
 857 C
 858 C
 859 C
 860 C
 861 C
 862 C
 863 C
 864 C
 865 C
 866 C
 867 C
 868 C
 869 C
 870 C
 871 C
 872 C
 873 C
 874 C
 875 C
 876 C
 877 C
 878 C
 879 C
 880 C
 881 C
 882 C
 883 C
 884 C
 885 C
 886 C
 887 C
 888 C
 889 C
 890 C
 891 C
 892 C
 893 C
 894 C
 895 C
 896 C
 897 C
 898 C
 899 C
 900 C
 901 C
 902 C
 903 C
 904 C
 905 C
 906 C
 907 C
 908 C
 909 C
 910 C
 911 C
 912 C
 913 C
 914 C
 915 C
 916 C
 917 C
 918 C
 919 C
 920 C
 921 C
 922 C
 923 C
 924 C
 925 C
 926 C
 927 C
 928 C
 929 C
 930 C
 931 C
 932 C
 933 C
 934 C
 935 C
 936 C
 937 C
 938 C
 939 C
 940 C
 941 C
 942 C
 943 C
 944 C
 945 C
 946 C
 947 C
 948 C
 949 C
 950 C
 951 C
 952 C
 953 C
 954 C
 955 C
 956 C
 957 C
 958 C
 959 C
 960 C
 961 C
 962 C
 963 C
 964 C
 965 C
 966 C
 967 C
 968 C
 969 C
 970 C
 971 C
 972 C
 973 C
 974 C
 975 C
 976 C
 977 C
 978 C
 979 C
 980 C
 981 C
 982 C
 983 C
 984 C
 985 C
 986 C
 987 C
 988 C
 989 C
 990 C
 991 C
 992 C
 993 C
 994 C
 995 C
 996 C
 997 C
 998 C
 999 C
 1000 C
 1001 C
 1002 C
 1003 C
 1004 C
 1005 C
 1006 C
 1007 C
 1008 C
 1009 C
 1010 C
 1011 C
 1012 C
 1013 C
 1014 C
 1015 C
 1016 C
 1017 C
 1018 C
 1019 C
 1020 C
 1021 C
 1022 C
 1023 C
 1024 C
 1025 C
 1026 C
 1027 C
 1028 C
 1029 C
 1030 C
 1031 C
 1032 C
 1033 C
 1034 C
 1035 C
 1036 C
 1037 C
 1038 C
 1039 C
 1040 C
 1041 C
 1042 C
 1043 C
 1044 C
 1045 C
 1046 C
 1047 C
 1048 C
 1049 C
 1050 C
 1051 C
 1052 C
 1053 C
 1054 C
 1055 C
 1056 C
 1057 C
 1058 C
 1059 C
 1060 C
 1061 C
 1062 C
 1063 C
 1064 C
 1065 C
 1066 C
 1067 C
 1068 C
 1069 C
 1070 C
 1071 C
 1072 C
 1073 C
 1074 C
 1075 C
 1076 C
 1077 C
 1078 C
 1079 C
 1080 C
 1081 C
 1082 C
 1083 C
 1084 C
 1085 C
 1086 C
 1087 C
 1088 C
 1089 C
 1090 C
 1091 C
 1092 C
 1093 C
 1094 C
 1095 C
 1096 C
 1097 C
 1098 C
 1099 C
 1100 C
 1101 C
 1102 C
 1103 C
 1104 C
 1105 C
 1106 C
 1107 C
 1108 C
 1109 C
 1110 C
 1111 C
 1112 C
 1113 C
 1114 C
 1115 C
 1116 C
 1117 C
 1118 C
 1119 C
 1120 C
 1121 C
 1122 C
 1123 C
 1124 C
 1125 C
 1126 C
 1127 C
 1128 C
 1129 C
 1130 C
 1131 C
 1132 C
 1133 C
 1134 C
 1135 C
 1136 C
 1137 C
 1138 C
 1139 C
 1140 C
 1141 C
 1142 C
 1143 C
 1144 C
 1145 C
 1146 C
 1147 C
 1148 C
 1149 C
 1150 C
 1151 C
 1152 C
 1153 C
 1154 C
 1155 C
 1156 C
 1157 C
 1158 C
 1159 C
 1160 C
 1161 C
 1162 C
 1163 C
 1164 C
 1165 C
 1166 C
 1167 C
 1168 C
 1169 C
 1170 C
 1171 C
 1172 C
 1173 C
 1174 C
 1175 C
 1176 C
 1177 C
 1178 C
 1179 C
 1180 C
 1181 C
 1182 C
 1183 C
 1184 C
 1185 C
 1186 C
 1187 C
 1188 C
 1189 C
 1190 C
 1191 C
 1192 C
 1193 C
 1194 C
 1195 C
 1196 C
 1197 C
 1198 C
 1199 C
 1200 C
 1201 C
 1202 C
 1203 C
 1204 C
 1205 C
 1206 C
 1207 C
 1208 C
 1209 C
 1210 C
 1211 C
 1212 C
 1213 C
 1214 C
 1215 C
 1216 C
 1217 C
 1218 C
 1219 C
 1220 C
 1221 C
 1222 C
 1223 C
 1224 C
 1225 C
 1226 C
 1227 C
 1228 C
 1229 C
 1230 C
 1231 C
 1232 C
 1233 C
 1234 C
 1235 C
 1236 C
 1237 C
 1238 C
 1239 C
 1240 C
 1241 C
 1242 C
 1243 C
 1244 C
 1245 C
 1246 C
 1247 C
 1248 C
 1249 C
 1250 C
 1251 C
 1252 C
 1253 C
 1254 C
 1255 C
 1256 C
 1257 C
 1258 C
 1259 C
 1260 C
 1261 C
 1262 C
 1263 C
 1264 C
 1265 C
 1266 C
 1267 C
 1268 C
 1269 C
 1270 C
 1271 C
 1272 C
 1273 C
 1274 C
 1275 C
 1276 C
 1277 C
 1278 C
 1279 C
 1280 C
 1281 C
 1282 C
 1283 C
 1284 C
 1285 C
 1286 C
 1287 C
 1288 C
 1289 C
 1290 C
 1291 C
 1292 C
 1293 C
 1294 C
 1295 C
 1296 C
 1297 C
 1298 C
 1299 C
 1300 C
 1301 C
 1302 C
 1303 C
 1304 C
 1305 C
 1306 C
 1307 C
 1308 C
 1309 C
 1310 C
 1311 C
 1312 C
 1313 C
 1314 C
 1315 C
 1316 C
 1317 C
 1318 C
 1319 C
 1320 C
 1321 C
 1322 C
 1323 C
 1324 C
 1325 C
 1326 C
 1327 C
 1328 C
 1329 C
 1330 C
 1331 C
 1332 C
 1333 C
 1334 C
 1335 C
 1336 C
 1337 C
 1338 C
 1339 C
 1340 C
 1341 C
 1342 C
 1343 C
 1344 C
 1345 C
 1346 C
 1347 C
 1348 C
 1349 C
 1350 C
 1351 C
 1352 C
 1353 C
 1354 C
 1355 C
 1356 C
 1357 C
 1358 C
 1359 C
 1360 C
 1361 C
 1362 C
 1363 C
 1364 C
 1365 C
 1366 C
 1367 C
 1368 C
 1369 C
 1370 C
 1371 C
 1372 C
 1373 C
 1374 C
 1375 C
 1376 C
 1377 C
 1378 C
 1379 C
 1380 C
 1381 C
 1382 C
 1383 C
 1384 C
 1385 C
 1386 C
 1387 C
 1388 C
 1389 C
 1390 C
 1391 C
 1392 C
 1393 C
 1394 C
 1395 C
 1396 C
 1397 C
 1398 C
 1399 C
 1400 C
 1401 C
 1402 C
 1403 C
 1404 C
 1405 C
 1406 C
 1407 C
 1408 C
 1409 C
 1410 C
 1411 C
 1412 C
 1413 C
 1414 C
 1415 C
 1416 C
 1417 C
 1418 C
 1419 C
 1420 C
 1421 C
 1422 C
 1423 C
 1424 C
 1425 C
 1426 C
 1427 C
 1428 C
 1429 C
 1430 C
 1431 C
 1432 C
 1433 C
 1434 C
 1435 C
 1436 C
 1437 C
 1438 C
 1439 C
 1440 C
 1441 C
 1442 C
 1443 C
 1444 C
 1445 C
 1446 C
 1447 C
 1448 C
 1449 C
 1450 C
 1451 C
 1452 C
 1453 C
 1454 C
 1455 C
 1456 C
 1457 C
 1458 C
 1459 C
 1460 C
 1461 C
 1462 C
 1463 C
 1464 C
 1465 C
 1466 C
 14

```
57      FORMAT (110,116X,14,12)
58      C
59      C      INITIALIZE SMOOTHER
60      C
61      DELTAT = 0.05
62      FLTOUT = 12
63      SMTOUT = 13
64      C
65      REWIND SMTOUT
66      C
67      C      REWIND FLTOUT
68      READ (FLTOUT,700) RUNID,MAXDAT,IC
69      WRITE (SMTOUT,700) RUNID,MAXDAT,IC
70      WRITE (6,600) RUNID
71      C
72      DO 0950 I = 1,20
73      SORT(1) = 1
74      CONTINUE
75      C
76      STATE TO MEASUREMENT TRANSITION MATRIX HKAL
77      C
78      DO 1500 I = 1,3
79      DO 1000 J = 1,9
80      HKAL(I,J) = 0.0
81      CONTINUE
82      1500
83      HKAL(1,1) = 1.0
84      HKAL(2,2) = 1.0
85      HKAL(3,3) = 1.0
86      C
87      STATE TRANSITION MATRIX PHI
88      C
89      DO 3000 I = 1,9
90      DO 2000 J = 1,9
91      PHI(I,J) = 0.0
92      CONTINUE
93      3000
94      DO 4000 I = 1,3
95      PHI(1,1) = 1.0
96      PHI(1+3,1+3) = 1.0
97      PHI(1+6,1+6) = 1.0
98      CONTINUE
99      4000
100     PHI(1,4) = DELTAT
101     PHI(2,5) = DELTAT
102     PHI(3,6) = DELTAT
103     PHI(1,7) = DELTAT**2 / 2.0
104     PHI(2,8) = DELTAT**2 / 2.0
105     PHI(3,9) = DELTAT**2 / 2.0
106     PHI(4,7) = DELTAT
107     PHI(5,8) = DELTAT
108     PHI(6,9) = DELTAT
109     C
110     C      LAMDA AND CAPLAM
111     C
112     DO 6000 I = 1,9
113     LAMDA(I) = 0.0
```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```
114      DO 5000 J = 1,9
115      CAPLAM(I,J) = 0.0
116      CONTINUE
117      5000 6000
118      C
119      C      BACKWARD SWEEP
120      C      DO 15000 NCOUNT = 1, MAXDAT
121      C
122      C
123      C      INPUT
124      C      ( FILTERED DATA MUST BE IN REVERSE CHRONOLOGICAL ORDER )
125      C
126      C      READ(FLTOUT,200) TIME,RESDUL(I),I=1,3
127      A      (BARN(I),I=7,9)
128      B      (SPSI(I),I=1,9)
129      C      ((PKAL(I,J),J=1,9),I=1,9)
130      D      ((QA(I,J),J=1,3),I=1,3)
131      E      ((RMEASI(I,J),J=1,3),I=1,3)
132      C
133      C      EXPAND BARN AND QA TO FULL DIMENSION
134      C
135      DO 7'00 I = 1,3
136      BARN(I) = 0.16666667*BARN(I+6)*DELTAT**2
137      BARN(I+3) = 0.5*BARN(I+6)*DELTAT
138      CONTINUE
139      C
140      DO 7400 I = 1,3
141      DO 7300 J = 1,3
142      QKAL(I,J) = QA(I,J)*DELTAT**4/36.
143      QKAL(I,J+3) = QA(I,J)*DELTAT**3/12.
144      QKAL(I,J+6) = QA(I,J) * DELTAT**2 / 6.0
145      QKAL(I+3,J) = QA(I,J)*DELTAT**3/12.
146      QKAL(I+3,J+3) = QA(I,J) * DELTAT**2 / 4.0
147      QKAL(I+3,J+6) = QA(I,J) * DELTAT / 2.0
148      QKAL(I+6,J) = QA(I,J) * DELTAT**2 / 6.0
149      QKAL(I+6,J+3) = QA(I,J) * DELTAT / 2.0
150      QKAL(I+6,J+6) = QA(I,J)
151      CONTINUE
152      7300 7400
153      C
154      C      COMPUTE SMOOTHED STATE
155      C
156      C      PKAL * PHI(TRANSPOSE)      STORE IN TEMP3
157      C
158      DO 9000 I = 1,9
159      DO 8000 J = 1,9
160      TEMP3(I,J) = 0.0
161      DO 7500 K = 1,9
162      TEMP3(I,J) = TEMP3(I,J) + PKAL(I,K) * PHI(J,K)
163      CONTINUE
164      8000 9000
165      CONTINUE
166      C
167      C      SMTSPS = SPS - TEMP3 * LAMDA --- SMOOTHED STATE
168      C      SMTBRN = BARN - QFLT * LAMDA --- SMOOTHED TRANSITION ERROR
169      C
170      DO 10000 I = 1,9
```

THIS PAGE IS BEST QUALITY PRACTICABLY
FROM COPY FURNISHED TO DDQ

```
171      SMTSPS(I) = SPS(I)
172      SMTBRN(I) = BARN(I)
173      DO 9500 J = 1,9
174      SMTSPS(I) = SMTSPS(I) - TEMP3(I,J)*LAMDA(J)
175      SMTBRN(I) = SMTBRN(I) - QKAL(I,J) * LAMDA(J)
176      CONTINUE
177      10000
178      C
179      C
180      C
181      C
182      C
183      COMPUTE SMOOTHED COVARIANCES
184      TEMP3 * CAPLAM * TEMP3(TRANSPOSE) STORE IN TEMP4
185      DO 10000 I = 1,9
186      DO 10500 J = 1,9
187      TEMP4(I,J) = 0.0
188      DO 10200 M = 1,9
189      DO 10100 K = 1,9
190      TEMP4(I,J) = TEMP4(I,J) + TEMP3(I,M) * CAPLAM(M,K) *
191      TEMP3(J,K)
192      CONTINUE
193      CONTINUE
194      CONTINUE
195      C
196      C
197      DO 12000 I = 1,9
198      DO 11500 J = 1,9
199      TEMP3(I,J) = 0.0
200      DO 11200 M = 1,9
201      DO 11100 K = 1,9
202      TEMP3(I,J) = TEMP3(I,J) + QKAL(I,J) * CAPLAM(M,K) *
203      QKAL(K,J)
204      CONTINUE
205      CONTINUE
206      CONTINUE
207      CONTINUE
208      DO 13000 I = 1,9
209      DO 12500 J = 1,9
210      SMTP = PKAL(I,J) - TEMP4(I,J) --- SMOOTHED STATE ERROR COVARIANCE
211      SMTQ = QKAL(I,J) - TEMP3(I,J) --- SMOOTHED PLANT ERROR COVARIANCE
212
213      DO 12000 I = 1,9
214      SMTP(I,J) = PKAL(I,J) - TEMP4(I,J)
215      SMTQ(I,J) = QKAL(I,J) - TEMP3(I,J)
216      CONTINUE
217      12500
218      CONTINUE
219      C
220      C
221      WRITE (SMTOUT,500) NCOUNT, TIME, NCOUNT, SORT(1)
222      *(SMTSPS(I), I=1,9), NCOUNT, SORT(2)
223      *((SMTP(I,J), J=1,9), NCOUNT, SORT(1+2), I=1,9)
224      *((SMTQ(I,J), J=1,9), I=7,9), NCOUNT, SORT(1+2)
225      *(SMTBRN(I), I=1,9), NCOUNT, SORT(13)
226      C
227      DO 1350 I = 1,3
```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```
228      1350      ADOT(I) = SMTRN(I+6) / DELTAT
229      C          CONTINUE
230      C          WRITE (6,300) NCOUNT, TIME, (SMTPS(I), I=1,9), (ADOT(I), I=1,3)
231      C
232      C          IF ( MOD((NCOUNT - 1),20) .NE. 0 ) GO TO 14000
233      C          WRITE (6,400) ((SMTP(I,J), J=1,9), I =1,9)
234      C          ((SMTQ(I,J), J=7,9), I =7,9)
235      C          CONTINUE
236      A          14000
237      C          CONTINUE
238      C          UPDATE LAMDA AND CAPLAM
239      C          CALL LAMDAS
240      C
241      C          15000      CONTINUE
242      C          ENDFILE SMOUT
243      C          15000      CONTINUE
244      C          STOP
245      C
246      C          END
247      C
```

Table A-3 Subroutine CHLSKI
SUBROUTINE CHLSKI

```
1 2 3 C CHOLESKI DECOMPOSITION ( MATRIX INVERSION )
4 5 C COMMON /KALIN2/ T1   T2   T3
6 6 C COMMON /MATINV/ Z   SZ
7 7 C
8 8 C
9 9 C      REAL T1(9,9)
10 10 C      F   T2(3,3)
11 11 C      G   T3(3,9)
12 12 C
13 13 C      DOUBLE PRECISION Z(3,3)
14 14 C      A   SZ(3)
15 15 C
16 16 DO 2000 I = 1,3
17 17 DO 1000 J = 1,3
18 18      Z(I,J) = T2(I,J)
19 19      CONTINUE
20 2000      CONTINUE
21 21      CALL DECOM
22 22      DO 5000 I = 1,9
23 23      DO 3000 J = 1,3
24 24      SZ(J) = T1(J,I)
25 25      CONTINUE
26 26      CALL SUBS
27 27      DO 4000 J = 1,3
28 28      T3(J,1) = SZ(J)
29 29      CONTINUE
30 30      5000      CONTINUE
31 31      C      RETURN
32 32      END
33 33
```

Table A-4 Subroutine CORECT

```

1      SUBROUTINE CORECT
2      C   UPDATE EXTRAPOLATION WITH MEASUREMENTS
3      C   COMMON /KAL10 / RESUL, BARN, SPS, PKAL
4      C   ,RMEAS ,HKAL ,QA ,RMEAS ,QKAL
5      A
6      C   COMMON /KALIN2/ T1 ,T2 ,T3
7      B
8      C   REAL RESUL (3)
9      C   A   'BARN(9)
10     C   'SPS(9)
11     C   B   'PKAL(9,9)
12     C   'SPS(9)
13     C   C   'QA(3,3)
14     C   D   'RMEAS1(3,3)
15     C   E   'QKAL(9,9)
16     C   F   'ZMEAS(3)
17     C   G   'HKAL(3,9)
18     C   H   'ZMEAS(3)
19     C   I   'PKAL(9,9)
20     C   J   'RMEAS(3,3)
21
22     C   REAL T1(9,9)
23     C   F   'T2(3,3)
24     C   G   'T3(3,9)
25
26     C   COMPUTE HKAL * PKAL      STORE IN T1
27     C
28     DO 3000 K = 1,3
29     DO 2000 I = 1,9
30     T1(K,I) = 0.0
31     DO 1000 J = 1,9
32     T1(K,I) = T1(K,I) + HKAL(K,J) * PKAL(J,I)
33     CONTINUE
34     CONTINUE
35     CONTINUE
36
37     C
38     C
39     C
40     C
41     DO 6000 K = 1,3
42     DO 5000 I = 1,3
43     T2(K,I) = RMEAS(K,I)
44     DO 4000 J = 1,9
45     T2(K,I) = T2(K,I) + T1(K,J) * HKAL(I,J)
46     CONTINUE
47     4000
48     5000
49     6000
50     C
51     C
52     C
53     C
54     C
55     C
56     C
      RMEAS + HKAL * PKAL * HKAL(TRANSPOSE)      STORE IN T2
      CHLSKI COMPUTES T2 INVERSE * T1      STORE IN T3
      CALL CHLSKI
      CORRECT PKAL ( NOTE PKAL SYMETRIC )
      DO 9000 K = 1,9

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```
57      DO 8000 I = 1,9
58      DO 7000 J = 1,3
59      PKAL(K,I) = PKAL(K,I) - T1(J,K) * T3(J,I)
60      CONTINUE
61      8000
62      9000
63      C
64      C      STABILIZE PKAL
65      C
66      DO 9700 I = 1,9
67      PKAL(I,I) = AMAX1(PKAL(I,I),0.000001)
68      DO 9500 J = 1,9
69      PKAL(J,I) = 0.5*(PKAL(I,J) + PKAL(J,I))
70      CONTINUE
71      9700
72      C
73      C      INNOVATION ( ZMEAS - HKAL * SPS )      STORE IN RESDUL
74      C
75      DO 11000 I = 1,3
76      RESDUL(I) = ZMEAS(I)
77      DO 10000 J = 1,9
78      RESDUL(I) = RESDUL(I) - HKAL(I,J) * SPS(J)
79      CONTINUE
80      10000
81      11000
82      C      CORRECT SPS ( T3 TRANSPOSE IS THE KALMAN GAIN )
83      C
84      DO 13000 I = 1,9
85      DO 12000 J = 1,3
86      SPS(I) = SPS(I) + T3(J,I) * RESDUL(J)
87      CONTINUE
88      12000
89      13000
90      C      RETURN
91      END
```

Table A-5 Subroutine CTRF

```

1      SUBROUTINE CTRF
2
3      C COMPUTES COORDINATE TRANSITION MATRIX, TRV, FROM REFERENCE TO FRENET FRAME
4      C AND FRENET FRAME ACCELERATION ERROR COVARIANCE - QAP
5
6      C      COMMON /KALIO / RESDUL,BARN ,SPS ,PKAL ,QA ,RMEAS1,QKAL
7      A      .2MEAS ,HKAL ,PHI ,RMEAS
8
9      C      COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
10     A      .TRS ,THRUST
11
12     C      COMMON /KALDAT/ PVAL ,DELTAT,CONE ,CTWO ,CONE ,SIGMR
13     A      .STDIAZ1,STDIAZ2,STDDEL1,STDDEL2
14     C      REAL   RESDUL(3)
15
16     A      .BARN(9)
17     B      .SPS(9)
18     C      .PKAL(9,9)
19     D      .QA(3,3)
20     E      .RMEASI(3,3)
21     F      .QKAL(9,9)
22     G      .2MEAS(3)
23     H      .HKAL(3,9)
24     I      .PHI(9,9)
25     J      .RMEAS(3,3)
26
27     C      REAL   QAP(3,3)
28     A      .VMAG
29     B      .TRV(3,3)
30     C      .ADOTVU
31     D      .AP
32     E      .GN
33     F      .RSPS
34     G      .TRS(3,3)
35     H      .THRUST
36
37     C      REAL   PVAL(9)
38     A      .DELTAT
39     B      .CONE
40     C      .CTWO
41     D      .CTHREE
42     E      .SIGMR
43     F      .STDIAZ1
44     G      .STDIAZ2
45     H      .STDDEL1
46     I      .STDDEL2
47
48     C      VMAG = SORT ( SPS(4)*SPS(4) + SPS(5)*SPS(5) + SPS(6)*SPS(6) )
49     C      VMAGP = SORT ( SPS(4)*SPS(4) + SPS(5)*SPS(5) )
50
51     C      CHECK FOR NEAR ZERO VELOCITY
52
53     C      IF ( VMAGP .LT. 50.0 ) GO TO 3000
54
55     C      FAST TARGET
56

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```
VMAGIN = 1.0 / VMAG
VMAGPI = 1.0 / VMAGPI
C
60      TRV(1,1) = SPS(4) * VMAGIN
       TRV(1,2) = SPS(5) * VMAGIN
       TRV(1,3) = SPS(6) * VMAGIN
C      COMPUTE NUMBER OF G'S
64      C
65      ADDOTVU = TRV(1,1)*SPS(7) + TRV(1,2)*SPS(8) + TRV(1,3)*SPS(9)
66      AN1 = SPS(7) - TRV(1,1) * ADDOTVU
67      AN2 = SPS(8) - TRV(1,2) * ADDOTVU
68      AN3 = SPS(9) - TRV(1,3) * ADDOTVU
69      GN2 = AN1*AN1 + AN2*AN2 + AN3*AN3
70      C
71      CHECK FOR NEAR ZERO ACCELERATION
72      C
73      IF ( GN2 .LT. 5.0 ) GO TO 1000
74      C
75      C
76      C
77      C
78      GN = SQRT ( GN2 )
79      TRV(2,1) = AN1 / GN
80      TRV(2,2) = AN2 / GN
81      TRV(2,3) = AN3 / GN
82      C
83      TRV(3,1) = TRV(1,2) * TRV(2,3) - TRV(2,2) * TRV(1,3)
84      TRV(3,2) = TRV(2,1) * TRV(1,3) - TRV(1,1) * TRV(2,3)
85      TRV(3,3) = TRV(1,1) * TRV(2,2) - TRV(1,2) * TRV(2,1)
86      C
87      C
88      C
89      C
90      THRUST = ADDOTVU + 9.8 * TRV(1,3)
91      APN = 9.8 * TRV(2,3) + GN
92      AP = SQRT( 96.0*TRV(3,3)**2 + APN**2 )
93      C
94      QAP(1,1) = ( CONE*DELTAT )**2
95      QAP(2,2) = ( CTWO*DELTAT )**2
96      QAP(3,3) = ( AMIN1(GN*CTHREE,30.0)*DELTAT )**2
97      C
98      GO TO 2000
99      1000
100      C
101      C
102      C
103      TRV(2,1) = - SPS(5) * VMAGPI
       TRV(2,2) = SPS(4) * VMAGPI
104      TRV(2,3) = 0.0
105      TRV(3,1) = - SPS(4)* SPS(6) * VMAGIN * VMAGPI
       TRV(3,2) = - SPS(5)* SPS(6) * VMAGIN * VMAGPI
       TRV(3,3) = VMAGP * VMAGIN
106      C
107      C
108      C
109      C
110      QAP(1,1) = ( CONE * DELTAT )**2
111      QAP(2,2) = ( CTWO * DELTAT )**2
112      QAP(3,3) = ( CTWO * DELTAT )**2
113      C
```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```
114      GN = 0.0
115      C
116      2000    CONTINUE
117      GO TO 4000
118      3000    CONTINUE
119      C
120      C      SLOW TARGET (VELOCITY SMALL)
121      C
122      DO 3500  I = 1,3
123      DO 3250  J = 1,3
124      TRV(I,J) = 0.0
125      QAP(I,J) = 0.0
126      CONTINUE
127      TRV(I,I) = 1.0
128      QAP(I,I) = (CONE*DELTAT)**2
129      CONTINUE
130      CONTINUE
131      C      RETURN
132      4000    END
133
```

Table A-6 Subroutine CTRS

```

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56
C COMPUTES TRANSITION MATRIX, TRS, FROM REFERENCE TO ESTIMATED L.O.S FRAME
C COMMON /KALIO / RESUL,BARN ,SPS ,PKAL ,QA ,RMEAS1,QKAL
A   .ZMEAS ,HKAL ,PHI ,RMEAS
C COMMON /KALUT / QAP ,VMAG ,TRV ,ADOTVU,AP ,GN ,RSPS
A   .TRS
C   REAL   RESUL(3)
A   .BARN(9)
A   .SPS(9)
B   .PKAL(9,9)
C   .QA(3,3)
D   .RMEAS1(3,3)
E   .QKAL(9,9)
F   .ZMEAS1(3)
G   .HKAL(3,9)
H   .PHI(9,9)
I   .RMEAS1(3,3)
C   REAL   QAP(3,3)
A   .VMAG
B   .TRV(3,3)
C   .ADOTVU
D   .AP
E   .GN
F   .RSPS
G   .TRS(3,3)
H   .THRUST
C   RSPS = SQRT ( SPS(1)**2 + SPS(2)**2 + SPS(3)**2 )
RSPSP = SQRT ( SPS(1)**2 + SPS(2)**2 )
C   CHECK FOR NEAR ZERO RANGE
C   IF ( RSPSP .LT. 5.0 ) GO TO 1000
C   RANGE OK
C   TRS('.',1) = SPS(1) / RSPS
C   TRS('.',2) = SPS(2) / RSPS
C   TRS('.',3) = SPS(3) / RSPS
C   TRS(2,1) = - SPS(2) / RSPSP
C   TRS(2,2) = SPS(1) / RSPSP
C   TRS(2,3) = 0.0
C   TRS(3,1) = - SPS(1)*SPS(3) / ( RSPS * RSPSP )
C   TRS(3,2) = - SPS(2)*SPS(3) / ( RSPS * RSPSP )
C   TRS(3,3) = RSPSP / RSPS
C   GO TO 4000
C   CONTINUE
C   RANGE SMALL
C   DO 3000 1 = 1,3

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

57
58 DD 2000 J = 1.3
59 TRS(I,J) = 0.0
60 CONTINUE
61 TRS(I,I) = 1.0
62 CONTINUE
63 CONTINUE
64 RETURN
65 C

Table A-7 Subroutine DECOM

```

1      SUBROUTINE DECOM
2      C DECOMPOSE ( MATRIX INVERSION )
3
4      C      COMMON /MATINV/ Z      ,SZ
5      C      DOUBLE PRECISION Z(3,3)
6      C      A      ,SZ(3)
7
8      C
9
10     DO 3000  I = 1,3
11     K = 1 - 1
12     DO 2000  J = J,3
13     IF ( K .EQ. 0 )  GO TO 1500
14     DO 1000  M = 1,K
15     Z(I,J) = Z(I,J) - Z(M,J) * Z(M,I)
16     CONTINUE
17     1000  CONTINUE
18     1500  IF ( (J-I) .EQ. 0 )  GO TO 1750
19     Z(I,J) = Z(I,J) / Z(I,I)
20     GO TO 1875
21     CONTINUE
22     1750  Z(I,J) = DSQRT( DABS(Z(I,J)) )
23     CONTINUE
24     1875  CONTINUE
25     2000  CONTINUE
26     3000  CONTINUE
27     C      RETURN
28

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

Table A-8 Subroutine EXTRAP
SUBROUTINE EXTRAP

```

1      SUBROUTINE EXTRAP
2
3      C EXTRAPOLATES STATE VECTOR AND STATE COVARIANCE MATRIX
4
5      COMMON /KAL10 / RESDUL,BARN,SPS,PKAL,QA
6      COMMON /ZMEAS ,HKAL,RMEAS
7      REAL  RESDUL(3)
8      REAL  BARN(9)
9      REAL  SPS(9)
10     REAL  PKAL(9,9)
11     REAL  QA(3,3)
12     REAL  RMEAS(3,3)
13     REAL  HKAL(9,9)
14     REAL  QKAL(9,9)
15     REAL  ZMEAS(3)
16     REAL  HKAL(3,9)
17     REAL  PHI(9,9)
18     REAL  RMEAS(3,3)
19
20     REAL  ST(9)
21     REAL  TEMP(9,9)
22
23     C PREDICT STATE VECTOR
24
25     DO 2000  I = 1,9
26     ST(I) = 0.0
27     DO 1000  J = 1,9
28     ST(I) = ST(I) + PHI(I,J) * SPS(J)
29     CONTINUE
30
31     DO 3000  I = 1,9
32     SPS(I) = ST(I)
33     CONTINUE
34     DO 3500  I = 1,9
35     SPS(I) = SPS(I) + BARN(I)
36     CONTINUE
37
38     C PKAL * PHI ( TRANSPOSE ) STORE IN TEMP
39
40     DO 6000  K = 1,9
41     DO 5000  I = 1,9
42     TEMP(K,I) = 0.0
43     DO 4000  J = 1,9
44     TEMP(K,I) = TEMP(K,I) + PKAL(K,J) * PHI(I,J)
45     CONTINUE
46     5000
47     6000
48
49     C PREDICT PKAL AS QKAL + PHI * PKAL * PHI(TRANSPOSE)
50
51     DO 9000  K = 1,9
52     DO 8000  I = 1,9
53     PKAL(K,I) = QKAL(K,I)
54     DO 7000  J = 1,9
55     PKAL(K,I) = PKAL(K,I) + PHI(K,J) * TEMP(J,I)
56     CONTINUE

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

57 8000
58 9000
59 C
60 C
61 RETURN
62 END

Table A-9 Subroutine FILTER

```
1 2
2 3 C KALMAN ESTIMATOR OF POSITION, VELOCITY, AND ACCELERATION
3 4 C
4 5 C COMPUTE TRANSITION ERROR COVARIANCE MATRIX QKAL
5 6 C AND TRANSITION ERROR MEAN BARN
6 7 C
7 8 CALL CTRF
8 9 CALL NBAR
9 10 CALL Q
10 11 C
11 12 C EXTRAPOLATE STATE
12 13 C
13 14 CALL EXTRAP
14 15 C
15 16 C COMPUTE MEASUREMENT ERROR COVARIANCE MATRIX
16 17 C
17 18 CALL CTRS
18 19 CALL R
19 20 C
20 21 C UPDATE STATE AND STATE ERROR COVARIANCE
21 22 C
22 23 CALL CORRECT
23 24 C
24 25 RETURN
25 26 END
```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

Table A-10 Subroutine INTFLT

```

1      SUBROUTINE INTFLT
2
3      C INITIALIZATION OF KALMAN ALGORITHMS
4
5      COMMON /KAL10 / RESDUL,BARN      'SPS      'PKAL      'QA      'RMEASI,PKAL
6      A      'ZMEAS,PKAL      'RMEAS
7      C
8      COMMON /KALUT / QAP      'VMAG      'ADOTVU,AP      'GN      'RSPS
9      A      'TRV      'THRUST
10     C
11     COMMON /KALDAT/ PVAL      'DELTAT,CONE      'CTWO      'CTHREE,SIGMR
12     A      'STD1,STD2,STD3,STD4
13     C
14     REAL   RESDUL(3)
15     A      'BARN(9)
16     B      'SPS(9)
17     C      'PKAL(9,9)
18     D      'QA(3,3)
19     E      'RMEASI(3,3)
20     F      'PKAL(9,9)
21     G      'ZMEAS(3)
22     H      'PKAL(3,9)
23     I      'PHI(9,9)
24     J      'RMEAS(3,3)
25
26     C      REAL   QAP(3,3)
27     A      'VMAG
28     B      'TRV(3,3)
29     C      'ADOTVU
30     D      'AP
31     E      'GN
32     F      'RSPS
33     G      'TRS(3,3)
34     H      'THRUST
35
36     C      REAL   PVAL(9)
37     A      'DELTAT
38     B      'CONE
39     C      'CTWO
40     D      'CTHREE
41     E      'SIGMR
42     F      'STD1
43     G      'STD2
44     H      'STD3
45     I      'STD4
46     C      C      C      NOMINAL VALUES
47
48
49     PVAL(1) = 9000000.0
50     PVAL(2) = 9000000.0
51     PVAL(3) = 4000000.0
52     PVAL(4) = 10000.0
53     PVAL(5) = 10000.0
54     PVAL(6) = 10000.0
55     PVAL(7) = 400.0
56     PVAL(8) = 400.0

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```
57      C
58      C      PVAL(9) = 400.0
59      C      CONE = 1.0
60      C      CTWO = 4.0
61      C      CTHREE = 1.0
62      C      SIGMR = 2.0
63      C      STDAZ1 = 1.0
64      C      STDAZ2 = 0.0005
65      C      STDEL1 = 1.0
66      C      STDEL2 = 0.0C95
67      C
68      C      INITIAL STATE VECTOR
69      C
70      C      DO 1000 I = 1,9
71      C      SPS(I) = 0.0
72      C      CONTINUE
73      C
74      C      INITIAL COVARIANCE MATRIX PKAL
75      C
76      C      DO 3000 I = 1,9
77      C      DO 2000 J = 1,9
78      C      PKAL(I,J) = 0.0
79      C      CONTINUE
80      C      CONTINUE
81      C      DO 4000 I = 1,9
82      C      PKAL(I,I) = PVAL(I)
83      C      CONTINUE
84      C
85      C      INITIAL AND CONSTANT MEASUREMENT TRANSITION MATRIX MKAL
86      C
87      C      DO 6000 I = 1,3
88      C      DO 5000 J = 1,9
89      C      MKAL(I,J) = 0.0
90      C      CONTINUE
91      C      CONTINUE
92      C      MKAL(1,1) = 1.0
93      C      MKAL(2,2) = 1.0
94      C      MKAL(3,3) = 1.0
95      C
96      C
97      C
98      C
99      C
100     C      INITIAL AND CONSTANT STATE TRANSITION MATRIX PHI
101     C      DO 7000 I = 1,9
102     C      DO 6000 J = 1,9
103     C      PHI(I,J) = 0.0
104     C      CONTINUE
105     C      DO 9000 I = 1,3
106     C      PHI(I,I) = 1.0
107     C      PHI(I+3,I+3) = 1.0
108     C      PHI(I+6,I+6) = 1.0
109     C      CONTINUE
110     C      PHI(1,4) = DELTAT
111     C      PHI(2,5) = DELTAT
112     C      PHI(3,6) = DELTAT
113     C      PHI(1,7) = DELTAT**2 / 2.0
```

114 PHI(2.8) = DELTAT**2 / 2.0
 115 PHI(3.9) = DELTAT**2 / 2.0
 116 PHI(4.7) = DELTAT
 117 PHI(5.8) = DELTAT
 118 PHI(6.9) = DELTAT
 119
 120 C ACCELERATION NOISE COVARIANCE MATRIX IN FRENET FRAME
 121 C
 122 DO 11000 I = 1,3
 123 DO 10000 J = 1,3
 124 QAP(1,J) = 0.0
 125 CONTINUE
 126 10000 11000 CONTINUE
 127 QAP(1,1) = (CONE*DELTAT)**2
 128 QAP(2,2) = (CTWO*DELTAT)**2
 129 QAP(3,3) = (CTHREE*DELTAT)**2
 130
 131 C INITIAL TRANSITION ERROR COVARIANCE MATRIX QKAL
 132 C
 133 DO 12000 I = 1,9
 134 DO 1*500 J = 1,9
 135 QKAL(I,J) = 0.0
 136 CONTINUE
 137 12000 11500 CONTINUE
 138 C
 139 DO 13000 I = 1,3
 140 QKAL(I+6,I+6) = QAP(2,2)
 141 CONTINUE
 142 C
 143 C INITIAL TRANSITION ERROR VECTOR BARN
 144 C
 145 DO 14000 I = 1,9
 146 BARN(I) = 0.0
 147 CONTINUE
 148 C RETURN
 149 C
 150 END

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

Table A-11 Subroutine LAMDAS

```

1
2      SUBROUTINE LAMDAS
3
4      C UPDATE LAMDA AND CAPLAM
5
6      COMMON /KALIO / RESDUL,BARN
7      A      'SPS      'PKAL      'QA
8      C      'RMEAS      'PHI      'RMEASI, QKAL
9
10     COMMON /AB10 / LAMDA      CAPLAM
11
12     DOUBLE PRECISION RESDUL(3)
13     A      BARN(9)
14     B      SPS(9)
15     C      PKAL(9,9)
16     D      QA(3,3)
17     E      RMEASI(3,3)
18     F      QKAL(9,9)
19     G      ZMEAS(3)
20     H      HKAL(3,9)
21     I      PHI(9,9)
22     J      RMEAS(3,3)
23
24     DOUBLE PRECISION LAMDA(9)
25     A      CAPLAM(9,9)
26
27     DOUBLE PRECISION SAB(9,9)
28     A      TEMAB1(9,9)
29     B      TEMAB2(9)
30     C      TEMAB3(9,9)
31     D      TAB3(9)
32     E      TAB4(9,9)
33     F      TAB5(9,9)
34
35     C      UPDATE LAMDA
36     C      SAB = HKAL(TRANSPOSE) * RMEASI * HKAL
37
38     DO 4000 I = 1,9
39     DO 3000 J = 1,9
40     SAB(I,J) = 0.0
41     DO 2000 M = 1,3
42     DC 1000 N = 1.3
43     SAB(I,J) = SAB(I,J) + HKAL(M,I) * RMEASI(M,N) *
44     HKAL(N,J)
45     A      CONTINUE
46     CONTINUE
47     CONTINUE
48     CONTINUE
49     C      COMPUTE I - PKAL * SAB STORE IN TEMAB1
50     C
51     C
52     DO 7000 I = 1,9
53     DO 6000 J = 1,9
54     TEMAB1(I,J) = 0.0
55     TEMAB3(I,J) = 0.0
56     DO 5000 K = 1,9

```

57
 58 TEMAB1(I,J) = TEMAB1(I,J) - PKAL(I,K) * SAB(K,J)
 59 CONTINUE
 60 6000 CONTINUE
 61 7000 CONTINUE
 62 DO 7500 I = 1,9
 63 TEMAB1(I,I) = TEMAB1(I,I) + 1.0
 64 TEMAB3(I,I) = TEMAB3(I,I) + 1.0
 65 CONTINUE
 66 C
 67 C PHI(TRANSPOSE) * LAMDA STORE IN TEMAB2
 68 DO 9000 I = 1,9
 69 TEMAB2(I) = 0.0
 70 DO 8000 J = 1,9
 71 TEMAB2(I) = TEMAB2(I) + PHI(J,I) * LAMDA(J)
 72 CONTINUE
 73 8000 CONTINUE
 74 9000 CONTINUE
 75 C
 76 C TEMAB2 - HKAL(TRANSPOSE) * RMEASI * RESDUL STORE IN TAB3
 77 DO 14000 I = 1,9
 78 TAB3(I) = TEMAB2(I)
 79 DO 13000 J = 1,3
 80 DO 12000 K = 1,3
 81 TAB3(I) = TAB3(I) - HKAL(J,I) * RMEASI(J,K) * RESDUL(K)
 82 CONTINUE
 83 12000 CONTINUE
 84 13000 CONTINUE
 85 14000 CONTINUE
 86 C
 87 C LAMDA = TEMAB1(TRANSPOSE) * TAB3
 88 DO 16000 I = 1,9
 89 LAMDA(I) = 0.0
 90 DO 15000 J = 1,9
 91 LAMDA(I) = LAMDA(I) + TEMAB1(J,I) * TAB3(I)
 92 CONTINUE
 93 15000 CONTINUE
 94 16000 CONTINUE
 95 C
 96 C UPDATE CAPLAM
 97 C
 98 C PHI * TEMAB1 STORE IN TAB4
 99 DO 18000 I = 1,9
 100 DO 17000 J = 1,9
 101 TAB4(I,J) = 0.0
 102 DO 16500 K = 1,9
 103 TAB4(I,J) = TAB4(I,J) + PHI(I,K) * TEMAB1(K,J)
 104 CONTINUE
 105 16500 CONTINUE
 106 17000 CONTINUE
 107 18000 CONTINUE
 108 C
 109 C TAB4(TRANSPOSE) * CAPLAM * TAB4 STORE IN TAB5
 110 DO 22000 I = 1,9
 111 DO 21000 J = 1,9
 112 TAB5(I,J) = 0.0
 113 DO 20000 M = 1,9

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```
114      DO 19000 N = 1,9
115      TAB5(I,J) = TAB5(I,J) + TAB4(M,I) * CAPLAM(M,N) *
116      A
117      19000      CONTINUE
118      20000      CONTINUE
119      21000      CONTINUE
120      22000      CONTINUE
121      C
122      C      CAPLAM = TAB5 + SAB * TEMAB1
123      C
124      DO 25000 I = 1,9
125      DO 24000 J = 1,9
126      CAPLAM(I,J) = TAB5(I,J)
127      DO 23000 K = 1,9
128      CAPLAM(I,J) = CAPLAM(I,J) + SAB(I,K) * TEMAB1(K,J)
129      CONTINUE
130      24000      CONTINUE
131      25000      CONTINUE
132      C
133      RETURN
134      END
```

Table A-12 Subroutine NBAR

```

1      SUBROUTINE NBAR
2      C COMPUTES STATE TRANSITION ERROR VECTOR BARN
3      C
4      C COMMON /KALIO/ RESUL,BARN   ,SPS   ,PKAL   ,QA   ,RMEAS, QKAL
5      C      .ZMEAS ,HKAL   ,PHI   ,RMEAS
6      C      .RMEAS1, QKAL
7      C COMMON /KALUT/ QAP   ,VMAG   ,TRV   ,ADOTVU,AP   ,GN   ,RSPS
8      C      .TRUS
9      C
10     C COMMON /KALDAT/ PVAL   ,DELTAT,CONE   ,CTWO   ,CONE, CTHREE, SIGMR
11     C      .STDZ1,STDZ2,STDZ1,STDZ2,STDZ1,STDZ2
12     C
13     C      REAL   RESUL(3)
14     C      A      BARN(9)
15     C      .SPS(9)
16     C      .PKAL(9,9)
17     C      .QA(3,3)
18     C      .RMEAS1(3,3)
19     C      .QKAL(9,9)
20     C      .ZMEAS(3)
21     C      .HKAL(3,9)
22     C      .PHI(9,9)
23     C      .RMEAS(3,3)
24     C
25     C      REAL   QAP(3,3)
26     C      A      VMAG   ,TRV(3,3)
27     C      .ADOTVU,AP   ,GN
28     C      .RSPS
29     C      .TRUS
30     C
31     C      REAL   PVAL(9)
32     C      A      DELTAT
33     C      .CONE
34     C      .CTWO
35     C      .CTHREE
36     C      .SIGMR
37     C      .STDZ1
38     C      .STDZ2
39     C      .STDZ1
40     C      .STDZ1
41     C      .STDZ2
42     C      .STDZ1
43     C      .STDZ1
44     C      .STDZ1
45     C      .STDZ2
46     C
47     C      IF ( VMAG .LT. 50.0 ) GO TO 1000
48     C      TBARN1 = -GN**2 / VMAG *DELTAT
49     C      TBARN2 = ADOTVU *GN /VMAG *DELTAT
50     C      TBARN3 = 0.0
51     C      GO TO 2000
52     C      CONTINUE
53     C      TBARN1 = 0.0
54     C      TBARN2 = 0.0
55     C      TBARN3 = 0.0
56     C      CONTINUE

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

57 C
58 C
59 C
60 C BARN(7) = TRV(1,1)*TBARN1 + TRV(2,1)*TBARN2 + TRV(3,1)*TBARN3
61 C BARN(8) = TRV(1,2)*TBARN1 + TRV(2,2)*TBARN2 + TRV(3,2)*TBARN3
62 C BARN(9) = TRV(1,3)*TBARN1 + TRV(2,3)*TBARN2 + TRV(3,3)*TBARN3
63 C
64 C BARN(4) = BARN(7) * 0.5 * DELTAT
65 C BARN(5) = BARN(8) * 0.5 * DELTAT
66 C BARN(6) = BARN(9) * 0.5 * DELTAT
67 C
68 C BARN(1) = BARN(7) * 0.1666667 * DELTAT**2
69 C BARN(2) = BARN(8) * 0.1666667 * DELTAT**2
70 C BARN(3) = BARN(9) * 0.1666667 * DELTAT**2
71 C
72 C RETURN
73 END

Table A-13 Subroutine Q

```

1      SUBROUTINE Q
2      C COMPUTES STATE TRANSITION ERROR COVARIANCE MATRIX QKAL
3
4      COMMON /KAL10/ RESUL,BARN,SPS,PKAL,QA,RMEAS1,OKAL
5      A   .ZMEAS,PKAL
6      C   .ZMEAS,PKAL
7      C   COMMON /KALUT/ QAP,VMAG,TRV,ADOTVU,AP,GN,RSPS
8      A   .QAP,VMAG,TRV,ADOTVU,AP,GN,RSPS
9      C   COMMON /KALDAT/ PVAL,DELTAT,CONE,CTWO,CTHREE,SIGMR
10     A   .PVAL,DELTAT,CONE,CTWO,CTHREE,SIGMR
11     .STDAZ1,STDAZ2,STDELL1,STDELL2
12
13     C   REAL  RESUL(3)
14     A   .BARN(9)
15     B   .SPS(9)
16     C   .PKAL(9,9)
17     D   .QA(3,3)
18     E   .RMEAS1(3,3)
19     F   .OKAL(9,9)
20     G   .ZMEAS(3)
21     H   .PKAL(3,9)
22     I   .PHI(9,9)
23     J   .RMEAS(3,3)
24
25     C   REAL  QAP(3,3)
26     A   .VMAG
27     B   .TRV(3,3)
28     C   .ADOTVU
29     D   .AP
30     E   .GN
31     F   .RSPS
32     G   .TRS(3,3)
33     H   .THRUS
34
35     C   REAL  TEMP(3,3)
36
37     C   REAL  PVAL(9)
38     A   .DELTAT
39     B   .CONE
40     C   .CTWO
41     D   .CTHREE
42     E   .SIGMR
43     F   .STDAZ1
44     G   .STDAZ2
45     H   .STDELL1
46     I   .STDELL2
47
48     C   TRANSFORM ACCELERATION COVARIANCE MATRIX (QAP) TO
49     C   REFERENCE FRAME (QA)
50     C   QAP * TRV      STORE IN TEMP
51
52     C   DO 2000  I = 1,3
53     C   DO 1000  J = 1,3
54     C   TEMP(I,J) = 0.0
55
56

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```
57      1000          QA(I,J) = 0.0
58      2000          CONTINUE
59          DO 5000  I = 1,3
60          DO 4000  K = 1,3
61          DO 3000  J = 1,3
62          TEMP(I,K) = TEMP(I,K) + QAP(I,J) * TRV(J,K)
63          CONTINUE
64          CONTINUE
65          CONTINUE
66          CONTINUE
67          C
68          C      TRV(TRANSPOSE) * TEMP      STORE IN QA
69          C
70          DO 8000  I = 1,3
71          DO 7000  K = 1,3
72          DO 6000  J = 1,3
73          QA(I,K) = QA(I,K) + TRV(J,I) * TEMP(J,K)
74          CONTINUE
75          CONTINUE
76          CONTINUE
77          C
78          C      TRANSITION ERROR COVARIANCE MATRIX QKAL
79          C
80          DO 10000 I = 1,3
81          DO 9000  J = 1,3
82          QKAL(I,J) = QA(I,J)*DELTAT**4/36.
83          QKAL(I,J+3) = QA(I,J)*DELTAT**3/12.
84          QKAL(I,J+6) = QA(I,J)*DELTAT**2/6.0
85          QKAL(I+3,J) = QA(I,J)*DELTAT**3/12.
86          QKAL(I+3,J+3) = QA(I,J)*DELTAT**2/4.0
87          QKAL(I+3,J+6) = QA(I,J)*DELTAT**2/2.0
88          QKAL(I+6,J) = QA(I,J)*DELTAT**2/6.0
89          QKAL(I+6,J+3) = QA(I,J)*DELTAT**2/2.0
90          QKAL(I+6,J+6) = QA(I,J)
91          CONTINUE
92          CONTINUE
93          C
94          RETURN
95          END
```

Table A-14 Subroutine R

```

1      SUBROUTINE R
2      C COMPUTES THE MEASUREMENT NOISE COVARIANCE MATRIX R
3      C R INVERSE IS COMPUTED FOR THE BACKWARD SWEEP SIMOTHER
4      C
5      COMMON /KAL10 / RESDUL,BARN   ,SPS   ,PKAL  ,QA   ,RMEASI,OKAL
6      A      ,ZMEAS ,HKAL   ,PHI   ,RMEAS
7      C      COMMON /KALDAT/ PVAL ,DELTAT,CONE  ,CTWO  ,CTHREE,SIGMR
8      A      .STDAZ1,STDAZ2,STDDEL1,STDDEL2
9      C      COMMON /KALUT / QAP  ,VMAG  ,TRV   ,ADOTVU,AP   ,GN   ,RSPS
10     A      .TRS   ,THRUST
11     C      REAL   RESDUL(3)
12     A      BARN(9)
13     B      SPS(9)
14     C      PKAL(9,9)
15     D      QA(3,3)
16     E      RMEASI(3,3)
17     F      OKAL(9,9)
18     G      ZMEAS(3)
19     H      HKAL(3,9)
20     I      PHI(9,9)
21     J      RMEAS(3,3)
22     K      QAP(3,3)
23     L      VMAG(3,3)
24     M      TRV(3,3)
25     N      ADOTVU
26     O      AP
27     P      GN
28     Q      RSPS
29     R      TRS(3,3)
30     S      THRUST
31     T      TEMP1(3,3)
32     U      RLOS(3,3)
33     V      RLOSS(3,3)
34     W      TEMP2(3,3)
35     X      PVAL(9)
36     Y      DELTAT
37     Z      CONE
38     A      CTWO
39     B      CTHREE
40     C      SIGMR
41     D      STDAZ1
42     E      STDAZ2
43     F      STDDEL1
44     G      STDDEL2
45     H      SIGMR*2
46     I      SIGMR*2
47     J      SIGMR*2
48     K      SIGMR*2
49     L      SIGMR*2
50     M      SIGMR*2
51     N      SIGMR*2
52     O      SIGMR*2
53     P      SIGMR*2
54     Q      SIGMR*2
55     R      SIGMR*2
56     S      SIGMR*2
      COMPUTE COVARIANCE MATRIX IN ESTIMATED L.O.S FRAME
      SIGRxs = SIGMR*2

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

THIS PAGE IS BEST QUALITY PRACTICABLY
FROM COPY FURNISHED TO DDC

```
57      SIGRS = (STDAL1)**2+ ( RSPS*STDAAZ2)**2
58      SIGRS = (STDAL1)**2+ ( RSPS*STDAL2)**2
59      DO 2000 I = 1,3
60      DO 1000 J = 1,3
61      RLOS(I,J) = 0.0
62      RMEAS(I,J) = 0.0
63      RMEAS(I,J) = 0.0
64      RMEAS(I,J) = 0.0
65      TEMP1(I,J) = 0.0
66      TEMP2(I,J) = 0.0
67      CONTINUE
68      2000 CONTINUE
69      RLOS(1,1) = SIGRXS
70      RLOS(2,2) = SIGRYS
71      RLOS(3,3) = SIGRZS
72      RLOS(1,1) = 1.0 / SIGRXS
73      RLOS(2,2) = 1.0 / SIGRYS
74      RLOS(3,3) = 1.0 / SIGRZS
75      C      TRANSFORM COVARIANCE MATRIX, RLOS, TO REFERENCE FRAME
76      C
77      C
78      C
79      C
80      C      RLOS * TRS, STORE IN TEMP1
81      DO 5000 I = 1,3
82      DO 4000 K = 1,3
83      DO 3000 J = 1,3
84      TEMP1(I,K) = TEMP1(I,J) * RLOS(I,J) * TRS(J,K)
85      TEMP2(I,K) = TEMP2(I,J) * RLOS(I,J) * TRS(J,K)
86      CONTINUE
87      4000 CONTINUE
88      5000 CONTINUE
89      C      TRS(TRANSPOSE) * TEMP1
90      C
91      C
92      C
93      C
94      C
95      C
96      C
97      C
98      C
99      C
100     C
101     C
102     C      RETURN
      END
```

Table A-15 Subroutine SUBS

```

1      SUBROUTINE SUBS
2      C
3      C  RECURSIVE INVERSION
4      C
5      COMMON /MATINV/ Z ,SZ
6      C
7      DOUBLE PRECISION Z(3,3)
8      A      ,SZ(3)
9      C
10     SZ(1) = SZ(1) / Z(1,1)
11     C
12     C  FORWARD SUBSTITUTION
13     C
14     DO 2000 I = 2,3
15     K = I - 1
16     DO 1000 J = 1,K
17     SZ(I) = SZ(I) - Z(J,I) * SZ(J)
18     CONTINUE
19     SZ(I) = SZ(I) / Z(I,I)
20     CONTINUE
21     SZ(3) = SZ(3) / Z(3,3)
22     DO 4000 I = 2,3
23     K = I - 1
24     J1 = 4 - I
25     C
26     C  BACK SUBSTITUTION
27     C
28     DO 3000 J = 1,K
29     JT = 4 - J
30     SZ(J1) = SZ(J1) - SZ(JT) * Z(J1, JT)
31     CONTINUE
32     SZ(J') = SZ(J1) / Z(J1, J1)
33     CONTINUE
34     C
35     RETURN
36

```

THIS PAGE IS BEST QUALITY PRACTICABLE
 FROM COPY FURNISHED TO DDC

Next page is blank

DISTRIBUTION LIST
TR-246

<u>No. of Copies</u>	<u>Organization</u>	<u>No. of Copies</u>	<u>Organization</u>
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCPA-S 5001 Eisenhower Avenue Alexandria, VA 22333	2	Chief Defense Logistics Studies Information Exchange US Army Logistics Management Center ATTN: DRXMC-D FT Lee, VA 23801
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDE-R 5001 Eisenhower Avenue Alexandria, VA 22333	1	HQDA (SAUS-OR) WASH DC 20310
		2	HQDA (DAMO-RQA) COL Campbell/LTC Solomon WASH DC 20310
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCBSI-AD 5001 Eisenhower Avenue Alexandria, VA 22333	2	Commander US Army Concepts Analysis Agency ATTN: MOCA-MR MOCA-ADL 8120 Woodmont Avenue Bethesda, MD 20014
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCBSI-OR 5001 Eisenhower Avenue Alexandria, VA 22333	1	Commander US Army Operational Test and Evaluation Agency ATTN: CSTE-STS-L 5600 Columbia Pike Falls Church, VA 22041
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCBSI-SE 5001 Eisenhower Avenue Alexandria, VA 22333	4	Commander US Army Armament Research and Development Command ATTN: DRDAR-SE DRDAR-LCS DRDAR-SCF DRDAR-SCS DRCPM-ADG Dover, NJ 07801
2	Commander Defense Documentation Center ATTN: TCA Cameron Station Alexandria, VA 22314	2	Commander US Army Aviation Research and Development Command ATTN: DRDAV-BCA DRDAV-BCE PO Box 209 St. Louis, MO 63166

DISTRIBUTION LIST (Continued)
TR-246

<u>No. of Copies</u>	<u>Organization</u>	<u>No. of Copies</u>	<u>Organization</u>
1	Director Research and Technology Laboratories ATTN: DAVDL Moffet Field, CA	1	Commander US Army Troop Support and Aviation Readiness Command ATTN: DRSTS-BA 4300 Goodfellow Boulevard St. Louis, MO 63120
1	Commander US Army Electronics Research and Development Command ATTN: DRDEL-AP-OA (B. Zarwyn) 2800 Powder Mill Road Adelphi, MD 20783	2	Commander US Army Combat Developments Experimentation Command ATTN: ATEC-PL-M ATEC-PL-TL FT Ord, CA 93941
8	Commander US Army Missile Research and Development Command ATTN: DRDMI-X DRDMI-DS DRDMI-CD DRDMI-CW DRDMI-RSI DRDMI-Y DRCPM-HEL DRCPM-ROC Redstone Arsenal, AL 35809	2	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-PL ATAA-TG White Sands Missile Range, NM 88002
1	Commander US Army Tank Automotive Research and Development Command ATTN: DRDTA-V Warren, MI 48090	1	Commandant US Army Air Defense School ATTN: ATSA-CD-A FT Bliss, TX 79916
1	Commander US Army Aviation Center and Fort Rucker ATTN: ATZQ-CD FT Rucker, AL 36362		
1	Commander US Army Armament Materiel Readiness Command ATTN: DRSAR-LEP-L (Tech Lib) Rock Island, IL 62199	1	President US Army Air Defense Board ATTN: ATZC FT Bliss, TX 79916
3	Commander US Army Missile Materiel Readiness Command ATTN: DRSMI-D DRCPM-CE DRCPM-HA Redstone Arsenal, AL 35809	2	Commander US Army White Sands Missile Range ATTN: STEWS-TE-ML STEPS-NR-PA White Sands Missile Range, NM 88002

DISTRIBUTION LIST (Continued)
TR-246

<u>No. of Copies</u>	<u>Organization</u>	<u>No. of Copies</u>	<u>Organization</u>
1	Commander US Army Combined Arms Combat Developments Activity ATTN: ATCA-CA Fort Leavenworth, KS 66027	1	Director Institute for Defense Analysis ATTN: Dr. J. Ross 400 Army Navy Drive Arlington, VA 22202
1	Commandant US Military Academy ATTN: COL John L. Palmer Dept. of Engineering West Point, NY 10996	1	United States Air Force Academy Department of Astronautics and Computer Science ATTN: DFACS/LTC E. J. Bauman USAF Academy, CO 80840
1	Commander Office of Naval Research ATTN: Mr. S.M. Selig 800 North Quincy Street Arlington, VA 22217	<u>Aberdeen Proving Ground</u>	
1	Director Special Projects Office Department of the Navy Washington, DC 20360	Dir, AMSAA ATTN: D.O'Neill, AWD H. Peaker, AWD J. Meredith, AWD R. Scheder, AWD H. Burke, CSD T. Perkins, CSD S. Johnson, MSD	
1	Commander Naval Weapons Center ATTN: Code 753 China Lake, CA 93557	Cdr, TECOM ATTN: DRSTE-CT-A DRSTE-CM-A DRSTE-AD-A DRSTE-AD-S DRSTE-AD-M	
1	AF/XOOFA Washington, DC 20330	Dir, HEL ATTN: DRXHE-SP	
1	AF/SAV Washington, DC 20330	Dir, BRL ATTN: DRDAR-BLB (H. Reed) DRDAR-BLB (P. Yip)	
1	AFSC (DOVS) Andrews AFB Washington, DC 20331	Dir, MTD ATTN: STEAP-MT-M	
1	TAWC/OA (Mr. J. Durrenberger) Eglin AFB, FL 32542		
1	AFWL (SA) Kirtland AFB, NM 87117		